SAMPLING AND RECOVERY ON PARAMETRIC MANIFOLDS

by

Qing Zou

A thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Applied Mathematical and Computational Sciences
in the Graduate College of
The University of Iowa

May 2021

Thesis Committee: Mathews Jacob, Thesis Supervisor

Xueyu Zhu

Weiyu Xu

Sanvesh Srivastava

Prashant Nagpal

To my Dad

ACKNOWLEDGEMENTS

I would like to thank the people who provided me with support, encouragement, and guidance in gaining my PHD. Please forgive me in advance if I have left anyone out, as countless people helped me during my study at The University of Iowa.

Foremost, I would thank my thesis advisor Dr. Prof. Mathews Jacob. All of the work in this thesis is done with the supervision of Mathews. I can certainly say that without his help, completing my PhD would never be a reality. My background is applied mathematics and before joining the Computational Biomedical Imaging Group (CBIG), I had no experience of signal processing, image processing and medical imaging. It is Mathews who taught me the knowledges about these interesting areas. I must thank his patience during our discussion. I am also impressed by his clarity, calibration, and passion in research.

Next, I would like to thank my thesis committee members: Drs. Xueyu Zhu, Weiyu Xu, Sanvesh Srivastava and Prashant Nagpal. It is my great honor and pleasure to have all of them serving on my dissertation committee. I also thank all of them for spending time understanding my research work and offering valuable suggestions.

Several other professors at the University of Iowa have had an impact on my work: Tong Li, Ionut Chifan and Weimin Han from the Department of Mathematics; Merry Mani and Stanley Kruger from the Department of Radiology. I am from the Applied Mathematics and Computational Sciences program and working in the Department of Electrical and Computer Engineering. Jumping between two depart-

ments gives me the opportunity to have more friends and I would like to thank them for their help both in my life and study: HanQin Cai, Yanqing Shen, Meng Zhou, Jeungeun Park, Xinwei Chen, Mingxiu Sui, Biao Ma, Fahim Ahmed Zaman, Jirong Yi. I also want to extend my thanks to my past and present CBIG lab-mates who have impact on my research: Hemant Kumar Aggarwal, Abdul Haseeb Ahmed, Gregory Ongie, Sunrita Poddar, Arvind Balachandrasekaran, Sampurna Biswas, Yasir Albaqqal, Aniket Pramanik, Stephen Siemonsma. Special thanks go to those volunteers and patients who are willing to do the MR scan and provide us valuable datasets so that I can perform my research.

I am also grateful to Mr. Yifu Wang, Mrs. Siying Li, Mrs. Judith Burkart, Mr. Steve Beisler and Mr. Mike Burkart who helped my family fit into the local culture in many ways.

Now, I would like to thank my wife Shuhua Shi. Her support, encouragement, quiet patience, and unwavering love were undeniably. I believe I would not be who I am today without her encouragement. I also acknowledge my parents and parents-in-law for their understanding and support during my graduate school. Lastly, I want to thank my daughter Sylvia Zou. Her coming to the world is one of the most meaningful gifts for me.

ABSTRACT

Recent studies show the great potential of using deep-learned image regularization priors in computational imaging in a variety of application areas including remote sensing, microscopy, and medical imaging. The obtained image quality is often superior to approaches that rely on union of subspaces (UoS) priors, which are either hand-crafted (e.g. wavelet sparsity) or shallow-learned (e.g.dictionary learning). However, the theoretical understanding of deep-learning based reconstruction algorithms is lagging behind UoS schemes. The main objective of this thesis is to introduce a union of surfaces framework, which models high-dimensional data as points on a union of low-dimensional surfaces or manifolds. We will develop novel sampling theoretic results and algorithmic tools for the learning of signals and functions on surfaces. The computational structure of these methods bear remarkable similarity to deep learning architectures, while being more amenable to theoretical analysis.

We first investigate the learning of a Union of Surfaces model from noisy and incomplete data. We develop novel algorithms with sampling guarantees to learn a union of surfaces representation from (a) few fully training samples, (b) and few partially observed samples. We introduce bounds on the approximation error in representing an arbitrary surface, and introduce algorithms that can learn the model from noisy and missing data.

Next, we consider the learning of multidimensional functions on Union of Surfaces. We introduce a novel framework for the representation of multidimensional

functions, when the domain is restricted to union of surfaces. Unlike conventional reproducing kernel Hilbert space setting, the representation only involves a sparse combination of training samples, whose number depends on the surface complexity. Motivated by deep architectures, we will investigate the factorization of complex functions into simpler ones, thus facilitating their compact representation and efficient learning.

Finally, we develop reconstruction algorithms, which combine the learned generative models as priors with the imaging physics. The framework is then used to enable free breathing and ungated multicontrast cardiac MRI reconstructions from highly undersampled measurements.

PUBLIC ABSTRACT

The last decade has witnessed extensive research on computational imaging, where faster and cheaper acquisition methods are combined with computational algorithms to dramatically improve spatial and temporal resolution of images, while reducing the cost of scan and hardware. The standard practice is to pose the recovery as an optimization problem, where the cost function is sum of a data consistency term and an image prior. Hand-crafted (e.g. sparsity in the wavelet domain) and shallow-learning (e.g dictionary learning, low-rank methods) priors have been extensively studied. Most of these methods rely on a union of subspaces signal representation, which assumes that the signal can be represented by the linear combination of vectors from union of subspaces. These approaches are now well-understood with theoretical guarantees, fast algorithms, and commercial products in several areas, including magnetic resonance imaging (MRI). However, recent empirical studies have shown the significcantly superior performance of non-linear deep architectures for recovery and inference in a wide range of application areas. The improved performance of these algorithms over union of subspaces schemes may be attributed to their ability of the associated prior terms to exploit the complex non-linear redundancies in the data. Unfortunately, current deep architectures do not enjoy a sound theoretical understanding compared to union of subspaces methods.

In this thesis, we develop a continuous-domain framework to exploit the non-linear dependencies in high-dimensional image data, with the focus of using it in

computational imaging applications. We model the data as points on a union of surfaces. The focus of this thesis is to bridge the gap between well-understood union of subspaces (compressed-sensing) frameworks and deep learning methods that offer great empirical performance. The proposed framework is then extended to yield a more efficient and theoretically-founded framework for MRI data with non-linear structure.

TABLE OF CONTENTS

LIST OF TABLES xii

LIST OF FIGURES xiii

CHAPTER

1 INTRODUCTION 1

1.1 Overview 1
1.2 Background 2
1.3 Recovery of union of surfaces from noisy and sparse data 3
1.4 Learning functions on union of surfaces 5
1.5 Model based computational imaging using union of surfaces prior 5
1.6 Aligned & jointly recovery of multi-slice data using union of surfaces model 7

2 RECOVERY OF UNION OF SURFACES FROM NOISY AND SPARSE DATA: THEORY AND ALGORITHMS 8

2.1 Introduction 8
2.2 Parametric surface representation 11
2.2.1 Band-limited surface representation 12
2.2.1.1 Relation of bandlimited representation with polynomials 13
2.2.1.2 Non-uniqueness of level-set representation 14
2.2.1.3 Minimal bandwidth representation of a surface . 15
2.2.1.4 Irreducible bandlimited surfaces 17
2.3 Lifting mapping and low-dimensional feature spaces 19
2.3.1 Band-limited surface representation 21
2.3.1.1 Irreducible surface with minimal lifting ($\Gamma = \Lambda$) 21
2.3.1.2 Irreducible surface with non-minimal lifting ($\Gamma \supset \Lambda$) 21
2.3.1.3 Union of irreducible surfaces with $\Gamma \supset \Lambda_i$ 23
2.4 Worst-case guarantees for curve recovery 23
2.4.1 Annihilation relations for points on the curve 23
2.4.2 Curve recovery from samples 23
2.4.3 Irreducible band-limited planar curve: sampling theorem . 24
2.4.4 Union of irreducible curves: sampling theorem 27
2.4.5 Curve recovery with unknown Fourier support 31
2.4.6 Recovery of arbitrary curves 35

2.4.7 Application of curve recovery in segmentation 37
2.5 High probability guarantees for surface recovery from samples . . 41
2.5.1 Sampling theorems 41
2.5.1.1 Case 1: Irreducible surfaces with minimal lifting 42
2.5.1.2 Case 2: Union of irreducible surfaces with minimal lifting 46
2.5.1.3 Case 3: Non-minimal lifting 47
2.5.2 Surface recovery algorithm for the non-minimal setting . . 50
2.6 Surface recovery from noisy samples 52
2.6.1 Point cloud denoising in 2D 56
2.6.2 Point cloud denoising in 3D 57
2.7 Discussion and Conclusion 58
2.8 Appendix 60
2.8.1 Proof of Lemma 4 60
2.8.2 Proof of Proposition 5 61
2.8.3 Proof of Proposition 6 61
2.8.4 Proof of Proposition 7 62
2.8.5 Proof of Proposition 8 62
2.8.6 Proof of Proposition 1 63
2.8.7 Proof of results in Section 2.5 65
2.8.7.1 Intersection of surfaces 66
2.8.7.2 Proof of Proposition 9 67
2.8.7.3 Proof of Proposition 11 68
2.8.7.4 Proof of Proposition 13 69
2.8.7.5 Proof of Proposition 14 70

3 LEARNING FUNCTIONS ON UNION OF SURFACES: LINKS TO NEURAL NETWORK 71

3.1 Introduction 71
3.2 Recovery of functions on surfaces 75
3.2.1 Compact representation of features using anchor points . 76
3.2.2 Representation and learning of functions 78
3.2.3 Efficient computation using *kernel trick* 81
3.2.4 Optimization of the anchor points and coefficients 84
3.3 Relation to neural networks 85
3.3.1 Task/function learning from input output pairs 86
3.3.2 Relation to auto-encoders 87
3.4 Illustration in denoising 90
3.5 Conclusion 94

4 MODEL BASED COMPUTATIONAL IMAGING USING UNION OF SURFACES PRIOR: DEEP GENERATIVE STORM MODEL 96

4.1 Introduction . . . 96
4.2 Background . . . 100
4.2.1 Dynamic MRI from undersampled data: problem setup . 100
4.2.2 Smooth manifold models for dynamic MRI . . . 100
4.2.3 Unsupervised learning using Deep Image Prior . . . 101
4.3 Deep generative SToRM model . . . 102
4.3.1 Generative model . . . 103
4.3.2 Distance/Network regularization . . . 105
4.3.3 Latent vector regularization penalty . . . 108
4.3.4 Proposed optimization criterion . . . 108
4.3.5 Strategies to reduce computational complexity . . . 109
4.3.5.1 Approximate data term for accelerated convergence 109
4.3.5.2 Progressive training-in-time . . . 110
4.4 Implementation details and datasets . . . 112
4.4.1 Structure of the generator . . . 112
4.4.2 Datasets . . . 113
4.4.3 Quality evaluation metric . . . 114
4.4.4 State-of-the-art methods for comparison . . . 115
4.4.5 Hyperparameter tuning . . . 115
4.5 Experiments and results . . . 116
4.5.1 Impact of different regularization terms . . . 116
4.5.2 Benefit of progressive training-in-time approach . . . 118
4.5.3 Impact of size of the network . . . 118
4.5.4 Comparison with the state-of-the-art methods . . . 119
4.6 Conclusion . . . 121

5 ALIGNED & JOINTLY RECOVERY OF MULTI-SLICE DATA USING UNION OF SURFACES PRIOR . . . 127

5.1 Introduction . . . 127
5.2 Methods . . . 132
5.2.1 Forward model . . . 132
5.2.2 Proposed approach . . . 132
5.2.3 Acquisition scheme . . . 135
5.2.4 Training approach . . . 136
5.2.5 Comparison with state-of-the-art methods . . . 137
5.3 Results . . . 138
5.3.1 Impact of K-L divergence penalty . . . 138
5.3.2 Comparisons with current manifold methods . . . 141
5.4 Discussion & Conclusions . . . 145

6 SUMMARY . . . 146

REFERENCES 147

LIST OF TABLES

Table

3.1 The PSNR (dB) of the denoised results for the two testing natural images with different noise level. 93

4.1 Architecture of the generator $\mathcal{G}_\theta$. $\ell(\mathbf{z})$ means the number of elements in each latent vector. 112

4.2 Quantitative comparisons based on six datasets: We used six datasets to obtain the average SER, PSNR, SSIM, Brisque score, and time used for reconstruction. 119

LIST OF FIGURES

Figure

2.1 Illustration the fertility of our level set representation model in 3D. The three examples show that our model is capable to capture the geometry of the shape even though the shape has complicated topologies, which demonstrated that the representation is not restrictive. 17

2.2 Illustration of the annihilation relations in 2D. We assume that the curve is the zero level set of a band-limited function $\psi(\mathbf{x})$, shown by the red function in the top left and the plane slicing the function gives us the level set of the function. The Fourier coefficients of ψ, denoted by $\mathbf{c}$, are support limited in Λ, denoted by the red square on the figure in the bottom right. Each point on the curve satisfies $\psi(\mathbf{x}_i) = 0$. Using the representation of the curve, we thus have $\mathbf{c}^T \phi_\Lambda(\mathbf{x}_i) = 0$. Note that $\phi_\Lambda(\mathbf{x}_i)$ is the exponential feature map of the point $\mathbf{x}_i$, whose dimension is specified by the cardinality of the set Λ. This means that the feature map will lift each point in the level set to a Λ dimensional subspace whose normal vector is specified by $\mathbf{c}$, as illustrated by the plane and the red vector $\mathbf{c}$ in the top right. Note that if more than one closed curve are presented, each curve will be lifted to a lower dimensional subspace in the feature space, as shown by the two lines in the plane, and the lower dimensional spaces will span the Λ dimensional subspace. 20

2.3 The non-minimal filter bandwidth Γ (green) is illustrated along with the minimal filter bandwidth Λ (red). The set $\Gamma \ominus \Lambda$ (blue) contains all indices at which Λ can be centered, while remaining inside Γ. 22

2.4 Illustration of Proposition 5: We consider a curve $C[\psi]$ given by $\psi(\mathbf{x})$, where $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is support limited to a 3×3 region, shown in (a). The theorem guarantees the perfect recovery will happen if we have no less than $(k_1 + k_2)^2 = 36$ samples. We first randomly chose 36 samples on the curve. Then from these 36 randomly chosen samples, we obtained (b), which gives us perfect recovery of the original curve. Furthermore, we mentioned that we do not require any constraint on the distribution of samples on the curve. In (c), we randomly chose 36 samples from the left half part of the curve and we got perfect recovery as well. In (d), 36 samples are randomly chosen from the right half of the curve. From (d), we saw that perfect recovery of the whole curve was also obtained. For each case, the average time required for the recovery is about 1.2 second. 26

2.5 Illustration of Proposition 6: We consider a curve $C[\psi]$ on the top right, where $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is support limited to a 5×5 region. The level set function is shown in the top left. We consider the recovery from different number of samples of $C[\psi]$, sampled randomly. The sampling locations are marked by red crosses. Note that the theory guarantees the recovery when the number of samples exceeds $(k_1 + k_2)^2 = 100$ samples. However, we observe good recovery of the curve around 50 samples. Note that our theoretical results are worst-case guarantees, and in practice fewer samples are sufficient for good recovery as seen from Fig, 2.7. On average, the computational time required for the recovery of the curve using 50 points is about 1.5 second. 28

2.6 Illustration of Proposition 6: We consider the same curve $C[\psi]$ as specified by Fig 2.5 (b), which is given by the union of two irreducible curves with bandwidth 3×3. So the bandwidth of $C[\psi]$ is 5×5. According to Proposition 6, we will need to have around 100 samples to recover $C[\psi]$ and each of the two irreducible curves need to satisfy with the sampling condition. As we noted in Fig 2.5, our results are worst-case guarantees. We observe that when we have 50 points and those points are uniformly sampled on the two irreducible curves, we can successfully recover the whole curve, as shown in (a). Now, if we put most of the samples on one of the irreducible curves, we cannot fully recover the curve, as illustrated in (b) and (c). This implies that the sampling condition on each of the irreducible factors is necessary in Proposition 6. 29

2.7 Effect of number of sampled points on perfect reconstruction. We randomly generated several curves with different bandwidth and number of sampled points, and recovered the curves from these samples. The success of reconstruction of the curves averaged over several trials are shown in the above phase transition plot, as a function of bandwidth and number of sampled entries. The color indicates the frequency of success; the color black indicates that the true curve cannot be recovered in any of the experiments, while the color white represents that the true curve is recovered in all the experiments. It is seen that perfect recovery occurs whenever we have $\geq (k_1 + k_2)^2$ samples, as indicated by our worst-case guarantees. However, we note that good recovery is observed whenever the number of samples exceed the degrees of freedom $k_1 \cdot k_2$ 30

2.8 Illustration of Propositions 7 & 8: We consider the recovery of the curve $C[\psi]$ as specified by Fig 2.5 (b), assuming unknown bandwidth. We overestimate the support Γ as 11x11, while the original support of $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is 5×5. According to Propositions 7& 8, when the number of samples exceed $(k_1+k_2)(l_1+l_2) = 220$, the matrix is low-rank. The first row shows the results by using 220 samples. We display the Fourier transforms of the three null-space functions of Φ_Γ in (a), (b) and (c). This approach of visualizing the null-space functions is similar to the approaches in [47, 91]. All of these functions are zero on the $\mathcal{C}[\psi]$, in addition to possessing several other zeros. The sum of squares function, denoted by (2.5.2) is shown on the right column, captures the common zeros, which specifies the curve $\mathcal{C}[\psi]$. We use the SOS function as a surrogate for the greatest common divisor of the null-space functions. Note that the bound in Proposition 7 is also a worst-case guarantee. In the second row, the curve $C[\psi]$ was sampled on 100 random sampling locations, denoted by the red crosses. We see that the curve can be recovered well using just 100 samples. The computational time used to specify the curve using SOS function in this experiment is about 1.6 second 32

2.9 Comparison of the proposed curve recovery scheme in Section 2.4.2 with the adaptation of [64] described in Section 2.4.6. The shape is randomly sampled on the points shown in the first column. The second column consists of the curves recovered using the level-set based algorithm, while the last column shows the ones by the proposed scheme. The computational time required for the level-set based algorithm is about 66 seconds whereas the computational time required for the proposed algorithm is only about 6.4 seconds using 1000 samples. 37

2.10 Illustration of edge based segmentation using the band-limited curve model using (2.40) and the comparisons with the segmentation method DRLSE introduced in [64]. The DLRSE scheme requires curve initialization, indicated by the green squares in the DLRSE results. The red curves in each case show the final curves. The parameters of the algorithms are optimized manually to yield the best results. The results show that the proposed scheme can provide similar segmentation as DLRSE, while it does not need initialization and is guaranteed to converge to global minimum. The ranks we choose here are 500 and 1200 for cells image and church image respectively. 39

2.11 Illustration of sensitivity of our proposed image segmentation algorithm to the rank. From the segmentation results, we see that when the rank is small, simpler segmentation curves will be obtained. When the rank is chosen to be too high, we will obtain over-segmentation result. Thus, the rank is a good surrogate for the complexity of the curve. 39

2.12 Illustration of Theorem 10 in 2D. The irreducible curve given by (a) is the original curve, which is obtained by the zero level set of a trigonometric polynomial whose bandwidth is 3×3. According to Theorem 10, we will need at least 8 samples to recover the curve. In (b), we randomly choose 7 samples (the red dots) on the original curve (the gray curve). The blue dashed curve shows the recovered curve from this 7 samples. Since the sampling condition is not satisfied, the recovery failed. In (c), we randomly choose 8 points (the red dots). From (c), we see that the blue dashed curve (recovered curve) overlaps the gray curve (the original curve), meaning that we recover the curve perfectly. In (d) - (f), we showed the original trigonometric polynomial, the polynomial obtained from 7 samples and the polynomial obtained from 8 samples. 44

2.13 Illustration of Theorem 10 in 3D. The irreducible surface given by (a) is the original surface, which is given by the zero level set of a trigonometric polynomial whose bandwidth is $3 \times 3 \times 3$. According to Theorem 10, we will need at least 26 samples to recover the surface. In (b), we randomly choose 25 samples (the blue dots) on the original surface (the gray part). The red surface is what we recovered from the 25 samples. Since the sampling condition is not satisfied, the recovery failed. In (c), we randomly choose 26 points (the blue dots). From (c), we see that the red surface (recovered surface) overlaps the gray surface (the original surface), meaning that we recover the surface perfectly. 45

2.14 Illustration of Theorem 12. The original curve (a) is given by the zero set of a reducible trigonometric polynomial with bandwidth 5×5, which is the product of two trigonometric polynomials with bandwidth 3×3. According to the sampling theorem, we totally need at least 24 samples and each of the components needs to be sampled for at least 8 samples. We first choose 7 samples (red dots) on the first component and 17 samples (red circles) on the second one. The gray curve in (b) is the original curve and the blue dashed curve is what we recovered from the $7 + 17 = 24$ samples. Since the sampling condition is not satisfied, the recovery failed. In (c), we choose 8 samples (red dots) on the first component and 16 samples (red circles) on the second one. From (c), we see that the gray curve (the original curve) overlaps the blue dashed curve (recovered curve), meaning that we recovered the curve successfully. In (d), we choose 17 samples on the first component and 7 samples on the other one. From (d), we see that the recovery is not successful. In (e), we have 16 samples on the first component and 8 samples on the second one. The original curve overlaps the recovered one. So we recovered it perfectly. Lastly, we choose 8 samples on each of the component and we failed to recover the curve as shown in (f). Note that the recovered curves pass through the samples in all cases. 48

2.15 Effect of number of sampled points on surfaces reconstruction error. We randomly generated several surfaces with different bandwidths and number of sampled points, and tried to recover the surfaces from these samples. The reconstruction errors of the surfaces averaged over several trials are shown in the above phase transition plot, as a function of bandwidth and number of sampled entries. the color black indicates that the true surfaces can be recovered in any of the experiments, while the color white represents that the true surfaces are not recovered in all the experiments. It is seen that we can almost recover the surfaces with $|\Lambda| = k_1 \cdot k_2 \cdot k_3$ samples. . . 50

2.16 Illustration of the sampling fashion for non-minimal bandwidth. We consider the curve as shown in (a), which is given by the zero level set of a trigonometric polynomial of bandwidth 5×5. We choose the non-minimal bandwidth Γ as 11×11. According to the sampling condition for non-minimal bandwidth, we sampled on 72 random locations. We randomly chose two null-space vectors for the feature matrix of the sampling set, which gave us functions (c) and (d). We can see that all of these functions have zeros on the original zero set, in addition to processing several other zeros. The sum of squares function is shown in (e), showing the common zeros, which specifies the original curve. 53

2.17 Comparison between proposed denoising algorithm (KLR) and Garph Laplacian Regularized denoising algorithm (GLR) introduced in [148]. . . 58

2.18 Illustration of the points cloud denoising algorithm and surface recovery algorithm with unknown bandwidth. The first row shows the samples drawn from three surfaces. Noise is added to the samples (see (d), (i), (n)). Then we use the proposed algorithm to denoise the points. The parameter λ in (2.48) is chosen as 1.4 for the denoising algorithm. The number of iterations for the denoising algorithm is 30. The surfaces that are recovered from noisy samples and denoised samples are also presented for comparison. The bandwidth was chosen as $31 \times 31 \times 31$ for all the experiments. 59

3.1 Illustration of the local representation of functions in 2D. We consider the local approximation of the band-limited function in (b) with a bandwidth of 13×13, living on the band-limited curve shown in (a). The bandwidth of the curve is 3×3. The curve is overlaid on the function in (b) in yellow. The restriction of the function to the vicinity of the curve is shown in (c). Our results suggest that the local function approximation requires $13^2 - 11^2 = 48$ anchor points. We randomly select the points on the curve, as shown in (d). The interpolation of the function values at these points yields the global function shown in (e). The restriction of the function to the curve in (f) shows that the approximation is good. 80

3.2 bandwidth of the set Λ with different q values. 82

3.3 Visualization of kernels in $\mathbb{R}^2$ and the non-linear function γ with some commonly used activation functions. 82

3.4 Computational structure of function evaluation. (a) corresponds to (3.6) to compute the band-limited multidimensional function $\mathbf{f}$ on $\mathcal{S}[\psi]$. The inner-product between the input vector $\mathbf{x}$ and the anchor templates on the surface are evaluated, followed by non-linear activation functions γ to obtain the coefficients $\alpha_i(\mathbf{x})$. These coefficients are operated with the fully connected linear layers $\mathbf{K}_{\mathbf{A}}^{\dagger}$ and $\mathbf{F}(\mathbf{A})$. The fully connected layers can be combined to obtain a single fully connected layer $\widetilde{\mathbf{F}}$. Note that this structure closely mimics a neural network with a single hidden layer. (b) uses an additional quadratic layer, which combines functions of a lower bandwidth to obtain a function of a higher bandwidth. 84

3.5 Illustration of the surface learning network using the curve in Fig. 2.16. (a) and (b) are the learned results. We compared the learned curve (blue curve) with the original curve (red curve) in (c). From which we see that the two curves are almost the same, indicating that the learned network performs well. 90

3.6 Comparison of our learned denoiser using the proposed activation function and the ReLU activation function. The testing results show that the denoising performance using the proposed activation function is comparable to the performance using ReLU. The eight rows in the figure correspond to the original images, the noisy images, the denoised images using the proposed one-layer network, the denoised images using one layer ReLU network, the denoised images using the proposed two-layer network, the denoised images using two-layer ReLU network, the denoised images using dictionary learning and the denoised images using non-local means. The averaged PSNR of the denoised images using the proposed one-layer network, one layer ReLU network, proposed two-layer network, two-layer ReLU network, dictionary learning and non-local means are 19.68 dB, 20.03 dB, 20.86 dB, 17.48 dB, 14.76 dB and 14.28 dB respectively. From the quantitative results, we can see that our proposed one-layer network performs comparable to the one-layer ReLU network. For the proposed two-layer network, the performance is getting better from both quantitative and visual points of view. For the two-layer ReLU network, visually the performance is better than that of the one-layer ReLU network. But the PSNR is getting worse. The main reason that causes the low PSNR for the two-layer ReLU network is the change of the pixel values on each hand-written digit. 92

3.7 Comparison of the proposed denoising algorithms on the image "Man" with $\sigma = 20$. 94

3.8 Comparison of the proposed denoising algorithms on the image "Lighthouse" with $\sigma = 20$. 94

4.1 Illustration of (a) analysis SToRM and (b) generative SToRM. Analysis SToRM considers a non-linear (e.g. exponential) lifting of the data. If the original points lie on a smooth manifold, the lifted points lie on a low-dimensional subspace. The analysis SToRM cost function in (4.5) is the sum of the fit of the recovered images to the undersampled measurements and the nuclear norm of the lifted points. A challenge with analysis SToRM is its high memory demand and the difficulty in adding spatial regularization. The proposed method models the images as the non-linear mapping $\mathcal{G}_\theta$ of some latent vectors $\mathbf{z}_i$, which lie in a very low-dimensional space. Note that the same generator is used to model all the images in the dataset. The number of parameters of the generator and the latent variables is around the size of a single image, which implies a highly compressed representation. In addition, the structure of the CNN offers spatial regularization as shown in DIP. The proposed algorithm in (4.13) estimates the parameters of the generator and the latent variables from the measured data. A distance regularization prior is added to the generator to ensure that nearby points in the latent subspace are mapped to nearby points on the manifold. Similarly, a temporal regularization prior is added to the latent variables. The optimization is performed using ADAM with batches of few images. 104

4.2 Illustration of the distance penalty. The length of the curve connecting the images corresponding to $\mathbf{z}_1$ and $\mathbf{z}_2$ depends on the Frobenius norm of the Jacobian of the mapping $\mathcal{G}$ as well as the Euclidean distance $\|\mathbf{z}_1 - \mathbf{z}_2\|^2$. We are interested in learning a mapping that preserves distances; we would like nearby points in the latent space to map to similar images. We hence use the norm of the Jacobian as the regularization prior, with the goal of preserving distances. 107

4.3 Illustration of the progressive training-in-time approach. In the first level of training, the k-space data of all the frames are binned into one and we try to solve for the average image in this level. Upon the convergence of the first step, the parameters and latent variables are transferred as the initialization of the second step. In the second level of training, we divide the k-space data into M groups and try to reconstruct the M average images. Following the convergence, we can move to the final level of training, where the parameters obtained in the second step and the linear interpolation of the latent vectors in the second step are chosen as the initializations of the final step of training. 111

4.4 Illustration of the impact of the regularization terms in the proposed scheme with $d = 24$. We considered three cases in the experiment: (1) using both regularizations, (2) using only latent regularization, and (3) using only network regularization; these correspond to the blue, orange, and yellow curves in (a). In (b), (c), and (d), we showed the learned latent vectors for the three cases. The visual and quantitative comparisons of the three cases are shown in (e). 122

4.5 Comparisons of the reconstruction performance with and without the progressive training-in-time strategy using $d = 40$. From the plot of SER vs. running time, we can see that the progressive training-in-time approach yields better results with much less running time comparing to the training without using progressive training-in-time. Two reconstructed frames near the end of systole and diastole using SToRM500, the proposed scheme with progressive training-in-time and the proposed scheme without using the progressive training-in-time are shown in the plot as well for comparison purposes. The average Brisque scores for SToRM500, the reconstruction with progressive training-in-time, and the reconstruction without progressive training-in-time are $36.4, 37.3$ and 39.1 respectively. . 123

4.6 Impact of network size on reconstruction performance. In the experiments, we chose $d = 8, 16, 24, 32, 40$ and 48 to investigate the reconstruction performance. We used 500 frames for SToRM reconstructions (SToRM500) as the reference for SER comparisons. For the investigation of the impact of network size on the reconstructions, we used 150 frames. The diastolic and systolic states and the temporal profiles are shown in the figure for each case. The Brisque scores and average SER are also reported. It is worth noting that when $d = 40$, the results are even less blurred than the SToRM500 results, even though only one-third of the data are used. . . . 124

4.7 Comparisons with the state-of-the-art methods. The first column of (a) corresponds to the reconstructions from 500 frames ($\sim$ 25s of acquisition time), while the rest of the columns are recovered from 150 frames ($\sim$ 7.5s of acquisition time). The top row of (a) corresponds to the diastole phase, while the third row is the diastole phase. The second row of (a) is an intermediate one. Fig. (b) corresponds to the time profiles of the reconstructions. We observe that the proposed ($d = 40$) reconstructions exhibit less blurring and fewer artifacts when compared to SToRM150 and competing methods. 125

4.8 Comparisons with the state-of-the-art methods. The first column of (a) corresponds to the reconstructions from 500 frames ($\sim$ 25s of acquisition time), while the rest of the columns are recovered from 150 frames ($\sim$ 7.5s of acquisition time). The top row of (a) corresponds to the diastole phase, while the third row is the diastole phase. The second row of (a) is an intermediate one. Fig. (b) corresponds to the time profiles of the reconstructions. We chose $d = 40$ for the proposed scheme. We observe that the proposed reconstructions appear less blurred when compared to the conventional schemes. 125

4.9 Illustration of the framework of the proposed scheme with $d = 40$. We plot the latent variables of 150 frames in a time series on the first dataset. We showed four different phases in the time series: systole in End-Expiration (E-E), systole in End-Inspiration (E-I), diastole in End-Expiration (E-E), and diastole in End-Inspiration (E-I). A thin green line surrounds the liver in the image frame to indicate the respiratory phase. The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors. 126

4.10 Illustration of the framework of the proposed scheme with $d = 40$. We plot the latent variables of 150 frames in a time series. We showed four different phases in the time series: systole in End-Expiration (E-E), systole in End-Inspiration (E-I), diastole in End-Expiration (E-E), and diastole in End-Inspiration (E-I). The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors. 126

5.1 Illustration of the proposed scheme on a dataset with three slices. The latent vectors of the i^{th} slice and time instant t, denoted by $\mathbf{z}_{i,t}$, are fed into the deep generative model $\mathcal{G}_\theta$, which generates the multi-slice image volume $\rho_{i,t} = \mathcal{G}_\theta[\mathbf{z}_{i,t}]$. The latent vectors $\mathbf{z}_{i,t}$ and the parameters θ of the generative model are learned jointly from the entire k-t space data $\mathbf{b}_{i,t}, \forall i, t$. The data consistency term in (5.3) specified by $\sum_i \sum_t \|\mathcal{A}_{it}(\rho_{i,t}) - \mathbf{b}_{i,t}\|^2$ is the sum of the errors between the measured k-t space data of each slice and the multi-channel measurements of the corresponding slices. For example, the operator $\mathcal{A}_{it}$ extracts the i^{th} slice from $\rho_{i,t}$ and evaluates its multi-channel Fourier transform, which is compared with the measurements $\mathbf{b}_{i,t}$. We additionally use regularization priors on the network and the latent parameters to make the reconstruction problem well posed. 130

5.2 Illustration of the impact of the K-L divergence penalty. We use four slices (slices 3-6) in the first dataset from the healthy volunteer to generate the results. In (a), we show the multi-slice reconstructions without using the K-L divergence penalty. The latent vectors corresponding to slice 3, which is shown in the plot at the bottom of slice 3, are fed into the generator to obtain the multi-slice reconstructions. Since the latent vectors in this case have different distributions, the reconstructions of slices 4, 5, and 6 are of bad image quality. In (b), we show the multi-slice reconstructions using the K-L divergence penalty. We feed the latent vectors corresponding to slice 3 into the generator. From the plots of the latent vectors, which are shown at the bottom of Fig. (b), we can see that the latent vectors of each slice have the same distribution, hence resulting in good reconstruction. 140

5.3 Illustration of the framework of the proposed scheme and comparison with existing methods. The experiments are based on the first dataset from the healthy volunteer, and 8 slices are used. We compare the proposed multi-slice generative manifold approach with the analysis SToRM and single-slice generative SToRM approaches. We use the SToRM reconstructions from the data of 500 frames (a-SToRM:500) as the reference for quantitative comparison. For the comparisons, we use the data of 150 frames for the reconstruction. From the reported average SER, shown at the bottom of figures (a) and (b), one can see that the proposed multi-slice generative manifold approach offers better reconstructions than the competing methods. We also plot the latent variables of 150 frames in time series for the proposed method. In this experiment, we feed the latent vectors corresponding to slice 8 to generate the multi-slice reconstruction. We showed four different phases for two different slices that are reconstructed in the time series: systole in end-expiration (E-E), systole in end-inspiration (E-I), diastole in E-E and diastole in E-I. The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors. 142

5.4 Illustration of the framework of the proposed scheme and comparison with existing methods. The experiments are based on the second dataset from the healthy volunteer, and 5 slices are used. We compare the proposed multi-slice generative manifold approach with the analysis SToRM and single-slice generative SToRM approaches. We use the SToRM reconstructions from the data of 500 frames (a-SToRM:500) as the reference for quantitative comparison. For the comparisons, we use the data of 150 frames for the reconstruction. From the reported average SER, shown at the bottom of figures (a) and (b), one can see that the proposed multi-slice generative manifold approach offers better reconstructions than the competing methods. We also plot the latent variables of 150 frames in time series for the proposed method. The first three slices in this dataset have the liver appearing in the field of view, but it never appears in the last two slices. Therefore, it is hard to determine the respiratory phases for the last two slices. In this experiment, we feed the latent vectors corresponding to slice 2 to generate the multi-slice reconstruction. We showed four different phases for slice 3 that are reconstructed in the time series and two phases for slice 4. The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors. 143

5.5 Illustration of the framework of the proposed scheme and comparison with existing methods. The experiments are based on the COPD dataset. We compared the proposed multi-slice generative manifold approach with the analysis SToRM and single-slice generative SToRM approaches. For the comparisons, we use the data of 150 frames for the reconstruction. We also compare the results with the analysis SToRM reconstructions from the data of 600 frames (a-SToRM:600). The BRISQUE score is used for quantitative comparison. The numbers at the bottom of figures (a) and (b) are the average BRISQUE scores. From the reported BRISQUE scores, one can see that the proposed multi-slice generative manifold approach offers better perceptual image quality than the competing methods. We also plot the latent variables of 150 frames in time series for the proposed method. In this experiment, we feed the latent vectors corresponding to slice 2 to generate the multi-slice reconstruction. We showed three different phases for two different slices that are reconstructed in the time series: diastole (first row), systole (third row), and intermediate phase (second row). The latent vectors corresponding to the three different phases are indicated in the plot of the latent vectors. 144

CHAPTER 1

INTRODUCTION

1.1 Overview

The last decade has witnessed extensive research on computational imaging, where faster and cheaper acquisition methods are combined with computational algorithms to dramatically improve spatial and temporal resolution of images, while reducing the cost of scan and hardware. The standard practice is to pose the recovery as an optimization problem, where the cost function is sum of a data consistency term and an image prior. Hand-crafted (e.g. sparsity in the wavelet domain) and shallow-learning (e.g dictionary learning, low-rank methods) priors have been extensively studied. Most of these methods rely on a union of subspaces (UoS) signal representation, which assumes that the signal can be represented by the linear combination of vectors from union of subspaces. These approaches are now well-understood with theoretical guarantees, fast algorithms, and commercial products in several areas, including magnetic resonance imaging (MRI). However, recent empirical studies have shown the significantly superior performance of non-linear deep architectures for recovery and inference in a wide range of application areas. The improved performance of these algorithms over union of subspaces schemes may be attributed to their ability of the associated prior terms to exploit the complex non-linear redundancies in the data. Unfortunately, current deep architectures do not enjoy a sound theoretical understanding compared to union of subspaces methods. Therefore, an improved theoretical understanding of deep learning priors and their usage in image

recovery is desirable to optimize them for improved performance.

The learning of continuous functions from the training data have been extensively studied in the context of reproducing kernel Hilbert spaces (RKHS). The functions are recovered as a linear combination of basis functions, whose number is equal to the number of training points; the computational structure resembles the single hidden-layer neural network, whose number of nodes is equal to the number of training points. The usage of sparse machines and random features have been introduced to reduce the number of basis functions. While they do not enjoy the optimality of the RKHS solution, their empirical performance is often superior. Note that these methods essentially rely on a low-rank approximation of the kernel matrix. But an improved understanding of when and why low-rank kernels yield improved performance is desirable.

1.2 Background

Machine learning methods are usually explained by the widely accepted manifold assumption: the probability distribution $p(\mathbf{x})$ of the high dimensional data is concentrated around a non-linear low-dimensional manifold $\mathcal{M}$. For instance, the encoder in an auto-encoder learns the non-linear mapping $E : \mathcal{M} \rightarrow \mathcal{L}$, where $\mathcal{L}$ is the latent space. The decoder $D : \mathcal{L} \rightarrow \mathcal{M}$ maps the data back to the original space, thus learning a one-to-one mapping between $\mathcal{M}$ and $\mathcal{L}$. The rectified linear unit (ReLU) based network relies on a piecewise linear model to represent E and D. Both of these relations imply that $\mathcal{M}$ is a smooth non-linear manifold or surface, parametrized by $\mathcal{L}$.

Manifold methods, which rely on the above assumption, have a long history in machine learning and signal processing for data visualization, feature extraction and etc.. A popular approach is to discretize the problem as a weighted data graph from the data samples, whose connectivity reflects the neighborhood structure on the manifold.

The recovery of continuous functions from their T discrete samples (training data) $(\mathbf{x}_i, y_i)$ is extensively studied in the context of RKHS. The celebrated representer theorem states that if k is a positive definite kernel and $\mathcal{K}$ is the associated RKHS, the solution to

$$f^* = \arg\min_f \sum_{i=1}^{T} ||f(\mathbf{x}_i) - y_i||^2 + \gamma ||f||_{\mathcal{K}}^2 \tag{1.1}$$

is given by $f(\mathbf{x}) = \sum_{i=1}^{T} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$. Note that the above solution mimics a one layer neural network, where the number of hidden nodes is the number of training points T. The challenge with these approaches is the very large T in applications. While sparsity prior can be used during the learning, the solution does not match the solution of (1.1). The approximation of the solution using $R << T$ random features, which amounts to a low-rank approximation of the kernel is considered in [105]. A surprising result is that the approximation often outperforms the true solution using R function evaluations. An improved understanding of when and why low-rank kernels yield good performance is desirable and it will be considered in this thesis.

1.3 Recovery of union of surfaces from noisy and sparse data

We develop theory and algorithms to learn a union of surfaces model from training data. We consider the learning from few clean training points as well as

training data corrupted by noise and missing entries.

For the 2D case, where we have a union of curves, we relied on a generalization of the Bezout's theorem to determine the minimum number of samples needed for the learning of union of curves. This part of the work is an extension of the past work on continuous-domain compressed sensing methods for image super-resolution [83–86]. Note that the results obtained from the Bezout's theorem give us the worst-case guarantee for the learning of union of curves. We generalize the results to high dimensions (dimension > 2). The extension of the planar results to high dimensional setting is not a straight-forward extension. The first challenging is that the direct extension of Bezout's Theorem to higher dimensions cannot be readily used in the higher dimensional setting. Another challenge with our 2-D result is the conservative nature of our worst-case bound.

These challenges can be overcome by seeking an average case result. Rather than relying on the worst-case setting in the planar case, we will approach the proof from a probabilistic setting. Specifically, the probability that randomly drawn points on a surface overlapping with a low-degree curve/surface is expected to be small. We are encouraged by similar theoretical results in the context of algebraic geometry, which provides high probability recovery whenever the number of measurements exceed the degrees of freedom.

In practice, usually the samples that we have are corrupted by noise. To improve the robustness to noise, we solve

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} ||\mathbf{X} - \mathbf{Y}||_F^2 + \lambda ||\Phi(\mathbf{X})||_*,$$

where $||\cdot||_*$ is the nuclear norm that is the surrogate for the rank. Such nuclear norm minimization schemes are widely used to improve the robustness of finite rate of innovation schemes. This approach yields good empirical recovery of surfaces from highly noisy data.

1.4 Learning functions on union of surfaces

We develop theory and algorithms for the efficient representation of functions, whose domain is restricted to union of surfaces. Unlike the conventional RKHS theory, the number of kernel evaluations does not depend on the number of training data points, but on the complexity (rank) of the union of surfaces. In addition to making the framework practical, the low-rank structure of the features enables us to introduce perfect recovery guarantees from finite number of labeled training points. We also exploit the property of exponentials described before to factorize complex functions as products of simpler functions, which translate to a simpler computational structure. Since the computational structure is essentially a one-layer neural network, we expect the theoretical tools to shed valuable insights on the trade-offs in deep learning architectures.

1.5 Model based computational imaging using union of surfaces prior

We use the learned union of surfaces generative prior in computational imaging applications. We develop a framework for dynamic MRI by modeling the images in the time series as points on a union of surfaces. The central hypothesis is that the time profiles of the dataset are smooth non-linear functions of a few latent variables (e.g. cardiac/respiratory phases and MR parameters) and hence can be modeled as

points on a union of surfaces, which can be learned from the highly undersampled k-t space data.

Specifically, we assume that the dynamic image volumes in the dataset are smooth non-linear functions of a few latent variables, i.e., $\mathbf{x}_t = \mathcal{G}_\theta(\mathbf{z}_t)$, where $\mathbf{z}_t$ are the latent vectors in a low-dimensional space. $\mathbf{x}_t$ is the t-th generated image frame in the time series. This explicit formulation implies that the image volumes lie on a smooth non-linear union of surfaces in a high-dimensional ambient space. The latent variables capture the differences between the images (e.g., cardiac phase, respiratory phase, contrast dynamics, subject motion). We model the $\mathcal{G}$ using a convolution nerual network (CNN), which offers a significantly compressed representation. The compact model proportionately reduces the number of measurements needed to recover the images. In addition, the compression also enables algorithms with much smaller memory footprint and computational complexity. We propose to jointly optimize for the network parameters θ and the latent vector $\mathbf{z}_t$ based on the given measurements. The smoothness of the surfaces generated by $\mathcal{G}_\theta(\mathbf{z})$ depends on the gradient of $\mathcal{G}_\theta$ with respect to its input. To enforce the learning of a smooth image surface, we regularize the norm of the Jacobian of the mapping $\|J_z\mathcal{G}_\theta\|^2$. Similarly, the images in the time series are expected to vary smoothly in time. Hence, we also use a Tikhonov smoothness penalty on the latent vectors $\mathbf{z}_t$ to further constrain the solutions. We use the ADAM optimizer with stochastic gradients, where random batches of $\mathbf{z}_i$ and $\mathbf{b}_i$ are chosen at iteration to determine the parameters. Unlike traditional CNN methods that need extensive fully-sampled training data, this approach recover the images

relying only on the highly undersampled k-t space data.

1.6 Aligned & jointly recovery of multi-slice data using union of surfaces model

Continuing to the work introduced in the last section, we further use the union of surfaces model for the aligned and jointly recovery of the multi-slice dynamic MRI data. We again model the images in the time series as points on a union of surfaces. We note that for the multi-slice data, the cardiac and respiratory motion during the acquisition of the different slices are different. So we will use different latent vectors for each slice, while the generator will be the same for all volumes. The parameters of the generator and the latent time-series for each slice are jointly learned from the measured data of all the slices. As mentioned above, the manifold approaches suffer from the high memory demand. While for the proposed scheme, the memory footprint of the algorithm is determined by the network parameters θ and the latent vectors $\mathbf{z}$, and hence is orders of magnitude smaller than that of manifold approaches.

CHAPTER 2

RECOVERY OF UNION OF SURFACES FROM NOISY AND SPARSE DATA: THEORY AND ALGORITHMS

2.1 Introduction

The recovery of surfaces from finite number of unorganized and noisy points is an important problem, with applications to computer vision [16, 53] and image processing [31, 51, 130]. This problem is fundamentally ill-posed because one can find infinite number of surfaces that pass through the measured points. In addition, the reconstruction is challenging due to surface topology (e.g. the shape of the surface), and the variation of topology with noise. Popular approaches for shape representation with arbitrary topology include (a) explicit representations using a mesh or graphs [9, 16], and (b) implicit level-set representations [4, 12, 18]. In the first scheme, the shape is constructed from the noisy points as a graph, where the nodes corresponding to adjacent data points are connected. In the second approach, level set functions are constructed from the points [128]. Several methods were introduced to account for noisy data, including spectral graph theory, Laplacian/curvature flow [25, 82]. All of these methods suffer from the inherent parametrization of the surface, which often depends on the sampling density.

The main focus of this chapter is to introduce a unified continuous domain theory for the recovery of union of surfaces. We assume that the points live on a surface, which is the zero level set of a bandlimited function ψ. This property enables us to express ψ as a finite linear combination of complex exponentials, where the

weights are specified by the vector $\mathbf{c}$. The bandwidth of ψ denoted by Λ is a measure of the complexity of the surface [83]. When ψ is irreducible, we term the surface as an irreducible surface. When ψ can be factorized into multiple irreducible factors, we obtain an union of irreducible surfaces; each of the irreducible factors correspond to a closed connected surfaces.

The function ψ vanishes at all points $\mathbf{x}$ on the surface; i.e, $\psi(\mathbf{x}) = 0$. This implies that the weighted linear combination of complex exponential features of the point $\exp\left(j2\pi\mathbf{k}^T\mathbf{x}\right); \mathbf{k} \in \Lambda$, weighted by $\mathbf{c}$, will vanish for all points on the surface. In particular, $\mathbf{c}$ is the normal vector to the complex exponential features of the points on the surface. We term this property as the annihilation relation, which suggests that the complex exponential maps of the points on the surface lie in a subspace, whose normal vector is $\mathbf{c}$. Thus, the non-linear exponential mapping transforms the non-linear surface structure to the familiar low-rank or subspace structure, which well-is studied in signal processing. When we have a union of irreducible surfaces, the samples from each one of the irreducible components lie on a subspace; the mapping transforms the complex structure to a union of subspaces structure [36]. The dimension of the subspace spanned by the feature maps is dependent on the bandwidth Λ, and hence the complexity of the surface.

We use the subspace structure of the feature vectors to recover the surface from a few measurements. Specifically, we identify the coefficient vector as the unit norm null space vector of the feature matrix, which is unique up to a scaling with magnitude one. We also introduce efficient strategies when the bandwidth of the

surface is unknown. Specifically, we show that when the support is overestimated, there exist multiple linearly independent filters that will annihilate the exponential maps; the common zeros, or equivalently the zero level set of the greatest common divisor of the filters, uniquely specifies the surface in this case.

When the support is overestimated, the feature matrix has multiple linearly independent null-space vectors, and hence is low-rank. We note that the Gram matrix of the exponential features correspond to a kernel matrix, which connects the band-limited surface model with widely used non-linear low-rank kernel methods [114]. When the surface samples are noisy, we rely on a nuclear norm minimization formulation to denoise the points. Specifically, we seek to find the denoised surface samples such that their feature vectors form a low-rank matrix. We use an iterative re-weighted algorithm to solve the above optimization problem, which alternates between the estimation of a weight matrix that approximates the null-space and a quadratic sub-problem to recover the data. We note that the iterative algorithm bears strong similarity to Laplacian/curvature flows used in graph denoising, which provides the connection between implicit level-set and explicit graph-based surface representations. One can also derive a graph Laplacian matrix from the weight matrix, which will facilitate the smoothing of signals that live on the nodes of the graph. This graph can be viewed as a discrete mesh approximation to the points that live on the surface. Our experiments show that the Laplacian matrix obtained by solving the proposed optimization algorithm is more representative of the graph structure than classical methods [10], especially when it is estimated from noisy data. This frame-

work reveals links between recent advances in superresolution theory [21, 85, 113], manifold smoothness based regularization, as well as graph signal processing [120].

This work has connections with Logan's results [67] for the recovery of 1-D band-limited functions from their zero crossings, as well as their extensions to 2-D [147]. The main challenge of these works is the extreme sensitivity of the band-limited function to the location of the zero-crossings, when no amplitude information of the signal is used [147]; this has prompted the use of additional information including multi-level contours [147] and multi-scale edges [71]. By contrast, we focus on the recovery of the surface itself, rather than the band-limited function, which is considerably simpler. Specifically, we propose to recover the surface as the zero level set of the sum of squares of all band-limited functions that satisfy the sampling conditions. In addition, unlike [147], our results are also valid for the union of irreducible surfaces. The proposed work is built upon our prior work on annihilation based super-resolution image recovery [76, 83–87] that has similarities to algebraic shape recovery [37] and the recent work by Ongie et al., which considered polynomial kernels [87]. Our main focus is to generalize [87] to shift invariant kernels, which are more widely used in applications.

2.2 Parametric surface representation

In this work, we use the level set representation to describe a (hyper-)surface. We model a (hyper-)surface $\mathcal{S}$ in $[0,1)^n; n \geq 2$ as the zero level set of a function ψ:

$$\mathcal{S}[\psi] = \{\mathbf{x} \in \mathbb{R}^n | \psi(\mathbf{x}) = 0\}. \tag{2.1}$$

For example, when $n = 2$, $\mathcal{S}$ is a (hyper-)surface of dimension 1, which is typically a curve. We note that the level set representation is widely used in image segmentation [64]. The normal practice in image segmentation is the non-parametric level set representation of a time-dependent evolution function ψ, which results in the PDE-driven models. Note that the initialization of these models affects the stability and the rate of convergence of the methods. So good initialization of level set functions is usually a requirement for good segmentation.

Several authors have recently proposed to represent the level set function ψ as a linear combination of basis functions $\varphi_{\mathbf{k}}(\mathbf{x})$ [13, 140]. These schemes argue that the reduced number of parameters translate to fast and efficient algorithms. Besides, we do not require the good initialization in this setting. Motivated by these schemes, we represent $\psi(\mathbf{x})$ as

$$\psi(\mathbf{x}) = \sum_{\mathbf{k}\in\Lambda} \mathbf{c_k}\ \varphi_{\mathbf{k}}(\mathbf{x}). \tag{2.2}$$

Since the level set function is the linear combination of some basis functions, we term the corresponding zero level set as parametric zero level set. We note that the surface properties would depend on the specific basis functions and will indeed decide the type of the kernel used in the algorithms in Section 2.6. We now provide some examples of parametric representations, depending on the choices of the basis functions.

2.2.1 Band-limited surface representation

We assume that the surface is within $[0, 1)^n$. A well-studied representation for support limited functions is the Fourier exponential basis, which is widely used

in digital image processing [92, 124, 155], biomedical image processing [88, 101, 125], and geophysics [106]. The level set function can be assumed to be band-limited [88], when ψ is expressed as a Fourier series:

$$\psi(\mathbf{x}) = \sum_{\mathbf{k}\in\Lambda} \mathbf{c}_{\mathbf{k}} \exp(j2\pi\mathbf{k}^T\mathbf{x}), \quad \mathbf{x} \in [0,1)^n. \tag{2.3}$$

In the above representation, the set Λ denotes the bandwidth of the Fourier coefficients $\mathbf{c} = \{\mathbf{c}_{\mathbf{k}} : \mathbf{k} \in \Lambda\}$; its cardinality $|\Lambda|$ is the number of free parameters in the surface representation. We refer to Λ as the Fourier support of ψ and we note that we always choose the support to be symmetric with respect to the origin. This choice is governed by the relation of this representation with polynomials, described in the next subsection. The extension of Λ governs the degree of the polynomial.

In this work, we focus on the Fourier series representation due to its key benefits including well-developed theoretical tools, fast algorithms such as fast Fourier transform, orthogonality, and the property that $|\exp(j2\pi\mathbf{k}^T\mathbf{x})| = 1$, which results in stable representations and also facilitate the theory.

2.2.1.1 Relation of bandlimited representation with polynomials

We also note that bandlimited representations (2.3) have an intimate relation with polynomials [88]. In particular, we note that one can transform the polynomial basis to an exponential one by the one-to-one mapping $\nu_i : [0,1) \to \{z \in \mathbb{C} : |z| = 1\}$:

$$\nu_i(x_i) = \exp(j2\pi x_i) =: z_i. \tag{2.4}$$

We will make use of this correspondence to study the properties of the zero sets of (2.3). With this transformation, the representation (2.3) simplifies to the complex polynomial denoted as $\mathcal{P}[\psi]$, which is of the form

$$\mathcal{P}[\varphi](\mathbf{z}) = \sum_{\mathbf{k}\in\Lambda} c_{\mathbf{k}} \prod_{i=1}^{n} z_i^{k_i}. \tag{2.5}$$

Since the mapping involves powers of z_i, where z_i are specified by the trigonometric mapping (2.4), we term the expansion in (2.3) as a trigonometric polynomial.

We note that the mapping $\nu = (\nu_1, \cdots, \nu_n)$ defined by (2.4) is a bijection from $[0,1)^n$ onto the complex unit torus $\mathbb{T}^n = \{(z_1, \cdots, z_n) : |z_i| = 1, i = 1, \cdots, n\}$. Hence,

$$\psi(\mathbf{x}) = 0 \quad \Leftrightarrow \quad \mathcal{P}[\psi][\mathbf{z}] = 0 \text{ on } \mathbb{T}^n, \quad \text{where} \quad z_i = \nu_i(x_i), \quad i = 1, \cdots, n, \tag{2.6}$$

which implies that there is a one-to-one correspondence between the zero sets of ψ and the zeros of $\mathcal{P}[\psi]$ on the unit torus. Accordingly, we can study the algebraic properties of trigonometric polynomials and their zero sets by studying their corresponding complex polynomials under the mapping ν.

2.2.1.2 Non-uniqueness of level-set representation

We first show that the level set representation of a surface in (2.3) may not be unique, when the bandwidth of the representation is larger than the minimal one required to represent the surface. We first note that the function $\psi(\mathbf{x})$ with bandwidth Λ in (2.3) can be expressed with a larger bandwidth $\Gamma \supset \Lambda$ by zero filling the additional Fourier coefficients:

$$\psi(\mathbf{x}) = \sum_{\mathbf{k}\in\Gamma} \tilde{\mathbf{c}}_{\mathbf{k}} \exp(j2\pi\mathbf{k}^T\mathbf{x}), \quad \mathbf{x} \in [0,1)^n, \tag{2.7}$$

where the coefficients set $\tilde{\mathbf{c}}$ is the zero-filled version of the vector $\mathbf{c}$, denoted by $\tilde{\mathbf{c}} \in \mathbb{C}^{|\Gamma|}$:

$$\tilde{\mathbf{c}}_{\mathbf{k}} = \begin{cases} \mathbf{c}_{\mathbf{k}} & \text{if } \mathbf{k} \in \Lambda \\ 0 & \text{else} \end{cases}. \tag{2.8}$$

We note that the representation of the surface by functions with the larger bandwidth Γ is not unique. In particular, any uniform shift of the coefficients in the Fourier domain corresponds to a phase multiplication in the space domain:

$$\varphi' = \varphi \cdot \exp(j2\pi \mathbf{k}_0^T \mathbf{x}); \quad \mathbf{k}_0 \in \Gamma \ominus \Lambda. \tag{2.9}$$

Since $|\exp(j2\pi \mathbf{k}_0^T \mathbf{x})| = 1, \forall \mathbf{x}$, we can see that the zero sets of φ' are identical to that of φ.

Because the exponentials $\exp(j2\pi \mathbf{k}_0^T \mathbf{x})$ are orthogonal to each other, the functions φ' that has the same zero set as φ lives in a subspace of dimension $\Gamma \ominus \Lambda$. Here, $\Gamma \ominus \Lambda$ denote the set of all valid uniform shifts $\mathbf{k}_0$ of Λ, denoted by $\Lambda + \mathbf{k}_0$, that are contained in Γ. We will introduce the set $\Gamma \ominus \Lambda$ with more details in §2.3.1.2.

2.2.1.3 Minimal bandwidth representation of a surface

We note from the previous section that the multiplication with the phase term in (15) corresponds to multiplying the trigonometric polynomial in (2.5) by $\mathbf{z}^{\mathbf{k}_0}$; the degree of the resulting trigonometric polynomial φ' will be greater than that of φ. In this section, we show that out of all these polynomials, the one with smallest degree is unique. More importantly, the bandwidth of the above minimal polynomial can be used as a measure of the complexity of the surface. Specifically, a more complex surface would correspond to a polynomial with a larger bandwidth.

The following result shows that for any given surface $\mathcal{S}$, there exists a unique level set function ψ, whose coefficient set $\{\mathbf{c_k} : \mathbf{k} \in \Lambda\}$ has the smallest bandwidth.

Proposition 1. *For every (hyper-)surface $\mathcal{S}$ given by the zero level set of* (2.7)*, there is a unique (up to scaling) minimal trigonometric polynomial ψ, which satisfies $\psi(\mathbf{x}) = 0; \forall \mathbf{x} \in \mathcal{S}$. Any other trigonometric polynomial ψ_1 that also satisfies $\psi_1(\mathbf{x}) = 0; \forall \mathbf{x} \in \mathcal{S}$ will have $BW(\psi_1) \supseteq BW(\psi)$. Here, $BW(\psi)$ denotes the bandwidth of the function ψ.*

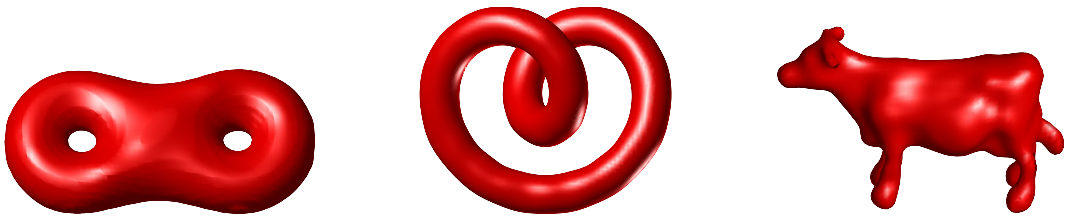
As seen from (15), the coefficients of ψ_1 can be the shifted version of the coefficients of ψ. Thus, the Fourier support of ψ_1 is larger than (contains) the Fourier support of ψ; the degree of the trigonometric polynomial ψ_1 is larger than the degree of the minimal polynomial ψ, which has the smallest degree or equivalently bandwidth. In this sense, the minimal polynomial ψ is unique, up to scaling. The proof of this result is given in Appendix 2.8.6. We refer to the ψ of the form (2.3) with the minimal bandwidth Λ that satisfy

$$\psi(\mathbf{x}) = 0; \quad \forall \mathbf{x} \in \mathcal{S} \tag{2.10}$$

as the *minimal trigonometric polynomial* of the surface $\mathcal{S}$.

In other words, when ψ is the minimal trigonometric polynomial of a surface $\mathcal{S}$, it does not have a factor with no zeros (i.e., never vanishes or vanishes only at isolated points on $[0, 1)^n$). In particular, if a polynomial has a factor with no zeros in $[0, 1)^n$, one can remove this factor and obtain a polynomial with a smaller bandwidth and with the same support set. Note from (2.5) that the minimal trigonometric polynomial will correspond to $\mathcal{P}[\psi]$ being a polynomial with the minimal degree.

As mentioned at the beginning of this section, the bandwidth Λ of the minimal polynomial of the surface $\mathcal{S}$ grows with the complexity of $\mathcal{S}$; a more oscillatory surface with a lot of details corresponds to a high bandwidth minimal polynomial, while a simple and highly smooth surface corresponds to a low bandwidth minimal polynomial. We hence consider $|\Lambda|$ as a *complexity measure* of the surface. Furthermore, we note that the surface model can approximate an arbitrary closed surface with any degree of accuracy, as long as the bandwidth is large enough [88]. One can refer to Fig.2 in [88] for illustration in 2D and see Fig. 2.1 for illustration in 3D. Here we illustrate this idea in 2D/3D for simplicity, but the approach is general for any dimensions.



(a) $17 \times 17 \times 17$ coeff. (b) $25 \times 25 \times 25$ coeff. (c) $33 \times 33 \times 33$ coeff.

Figure 2.1. Illustration the fertility of our level set representation model in 3D. The three examples show that our model is capable to capture the geometry of the shape even though the shape has complicated topologies, which demonstrated that the representation is not restrictive.

2.2.1.4 Irreducible bandlimited surfaces

We now introduce the concept of irreducible polynomials, which is important for our results. We term a surface to be irreducible if its minimal trigonometric

polynomial is irreducible. A polynomial is irreducible if it cannot be factorized into smaller factors, whose zero sets are within $[0,1)^n$. Most of the irreducible surfaces are simply connected (i.e., consist of a single connected component [1]). Intuitively, a general surface may be composed of several connected components, where each connected component is irreducible. In this case, we term the above surface as the union or irreducible surfaces. The minimal polynomial of the union of irreducible surfaces will be the product of the irreducible minimal polynomials of the individual connected components. The following definitions puts the above explanations into more concrete terms:

Definition 2. *A surface is termed as irreducible, if it is the zero set of an irreducible trigonometric polynomial.*

Definition 3. *A trigonometric polynomial $\psi(\mathbf{x})$ is said to be irreducible, if the corresponding polynomial $\mathcal{P}[\psi]$ is irreducible in $\mathbb{C}[z_1, \cdots, z_n]$. A polynomial p is irreducible over a field of complex numbers, if it cannot be expressed as the product of two or more non-constant polynomials with complex coefficients.*

When ψ can be written as the product of several irreducible components $\psi = \prod_{i=1}^{m} \psi_i$, then $\mathcal{S}[\psi]$ is essentially the union of irreducible surfaces:

$$\mathcal{S}[\psi] = \bigcup_{i=1}^{m} \mathcal{S}[\psi_i]. \tag{2.11}$$

[1] One can come up with counter examples of irreducible polynomials with multiple components. In this work, one can ignore these pathological counter examples and assume that an irreducible bandlimited surface will consist of only one connected component.

2.3 Lifting mapping and low-dimensional feature spaces

In this section, we show that there exists a non-linear transformation, which maps the points on an irreducible surface to a low-dimensional subspace. The transformation is intimately tied in with the specific choice of basis functions used to represent the surface. Our results show that the dimension of the subspace depends on the complexity of the surface, or equivalently the bandwidth of the minimal polynomial. We can use the rank of the feature matrix as a surrogate of the complexity of the surface to recover it, much like sparsity is used to recover signals in compressed sensing.

Consider the non-linear lifting mapping $\Phi_\Gamma : [0,1]^n \to \mathbb{C}^{|\Gamma|}$, obtained by evaluating the basis functions at $\mathbf{x}$:

$$\Phi_\Gamma(\mathbf{x}) = \begin{bmatrix} \varphi_{\mathbf{k}_1}(\mathbf{x}) \\ \vdots \\ \varphi_{\mathbf{k}_{|\Gamma|}}(\mathbf{x})) \end{bmatrix}. \tag{2.12}$$

We can view $\Phi_\Gamma(\mathbf{x})$ as the feature vector of the point $\mathbf{x}$, analogous to the ones used in kernel methods [114]. Here, $|\Gamma|$ denotes the cardinality of the set Γ. We denote the set

$$\mathcal{V}_\Gamma(\mathcal{S}) = \{\Phi_\Gamma(\mathbf{x}) | \mathbf{x} \in \mathcal{S}\} \tag{2.13}$$

as the feature space of the surface $\mathcal{S}$. Since any point on a surface $\mathcal{S}$ satisfies (2.1), the feature vectors of points from $\mathcal{S}$ satisfy

$$\mathbf{c}^T \Phi_\Gamma(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \mathcal{S}, \tag{2.14}$$

where $\mathbf{c}$ is the coefficients vector in the representation of ψ in (2.3). The above relation is illustrated in Fig. 2.2.

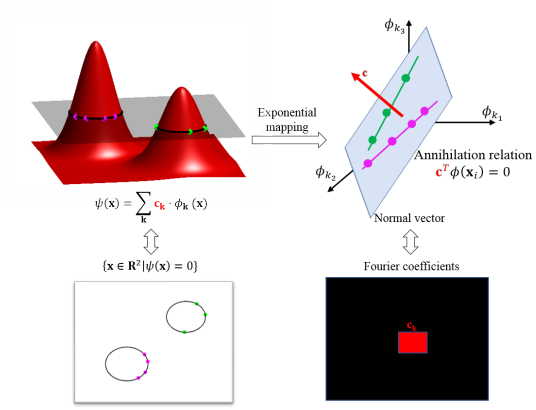


Figure 2.2. Illustration of the annihilation relations in 2D. We assume that the curve is the zero level set of a band-limited function $\psi(\mathbf{x})$, shown by the red function in the top left and the plane slicing the function gives us the level set of the function. The Fourier coefficients of ψ, denoted by $\mathbf{c}$, are support limited in Λ, denoted by the red square on the figure in the bottom right. Each point on the curve satisfies $\psi(\mathbf{x}_i) = 0$. Using the representation of the curve, we thus have $\mathbf{c}^T\phi_\Lambda(\mathbf{x}_i) = 0$. Note that $\phi_\Lambda(\mathbf{x}_i)$ is the exponential feature map of the point $\mathbf{x}_i$, whose dimension is specified by the cardinality of the set Λ. This means that the feature map will lift each point in the level set to a Λ dimensional subspace whose normal vector is specified by $\mathbf{c}$, as illustrated by the plane and the red vector $\mathbf{c}$ in the top right. Note that if more than one closed curve are presented, each curve will be lifted to a lower dimensional subspace in the feature space, as shown by the two lines in the plane, and the lower dimensional spaces will span the Λ dimensional subspace.

The relation (2.20) also implies that $\mathbf{c}$ is orthogonal to all the feature vectors of points living on $\mathcal{S}$ and hence a feature matrix constructed from points on the surface is rank deficient by one; i.e., the dimension of the feature space is at most $|\Gamma| - 1$. However, we now show that the feature matrix is often significantly low-rank depending on the geometry of the surface and the specific representations of the surface.

2.3.1 Band-limited surface representation

We now consider the case of an arbitrary point $\mathbf{x}$ on the zero level set of $\psi(\mathbf{x})$ with bandwidth Λ. Using (2.7), the lifting is specified by:

$$\Phi_\Lambda(\mathbf{x}) = \begin{bmatrix} \exp(j2\pi\mathbf{k}_1^T\mathbf{x}) \\ \exp(j2\pi\mathbf{k}_2^T\mathbf{x}) \\ \vdots \\ \exp(j2\pi\mathbf{k}_{|\Lambda|}^T\mathbf{x}) \end{bmatrix}. \tag{2.15}$$

We note from (2.15) that the lifting Φ can be evaluated with a larger bandwidth $\Gamma \supset \Lambda$. When the lifting is performed with the minimal bandwidth (i.e., $\Gamma = \Lambda$), we term the corresponding lifting as the *minimal lifting*.

We now analyze the dimension of the feature space $\mathcal{V}_\Lambda(\mathcal{S})$ for the minimal ($\Gamma = \Lambda$) and non-minimal lifting ($\Lambda \subset \Gamma$) cases. In both cases, we will show that the feature space is low-dimensional and is a subspace of $\mathbb{C}^{|\Lambda|}$.

2.3.1.1 Irreducible surface with minimal lifting ($\Gamma = \Lambda$)

We first focus on the case where ψ is an irreducible trigonometric polynomial and the bandwidth of the lifting is specified by Λ, which is the bandwidth of the minimal polynomial. The annihilation relation (2.20) implies that $\mathbf{c}$ is orthogonal to the feature vectors $\Phi_\Lambda(\mathbf{x})$. This implies that

$$\dim(\mathcal{V}_\Lambda) \leq |\Lambda| - 1. \tag{2.16}$$

2.3.1.2 Irreducible surface with non-minimal lifting ($\Gamma \supset \Lambda$)

We now consider the setting where the non-linear lifting is specified by $\Phi_\Gamma(\mathbf{x})$, where $\Lambda \subset \Gamma$. Because of the annihilation relation, we have

$$\tilde{\mathbf{c}}^T\ \Phi_\Gamma(\mathbf{x}) = 0,$$

where $\tilde{\mathbf{c}}$ is the zero filled coefficients in (2.7). Since the zero set of the function $\psi_{\mathbf{k}_0}(\mathbf{x}) = \psi(\mathbf{x}) \cdot \exp(j2\pi \mathbf{k}_0^T \mathbf{x})$ is exactly the same as that of ψ, we have

$$\sum_{\mathbf{k}} \mathbf{c}_{\mathbf{k}-\mathbf{k_0}} \exp(j2\pi \mathbf{k}^T \mathbf{x}) = 0; \quad \forall \mathbf{x} \in \mathcal{S}[\psi]. \tag{2.17}$$

This implies that any shift of $\tilde{\mathbf{c}}$ within $\Gamma \ominus \Lambda$, denoted by $\tilde{\mathbf{d}}_{\mathbf{k}} = \mathbf{c}_{\mathbf{k}-\mathbf{k}_0}$ will satisfy $\tilde{\mathbf{d}}^T \ \Phi_\Gamma(\mathbf{x}) = 0$. It is straightforward to see that $\tilde{\mathbf{d}}$ and $\tilde{\mathbf{c}}$ are linearly independent for all values of $\mathbf{k}_0$. We denote the number of possible shifts such that the shifted set $\Lambda + \mathbf{k}_0$ is still within Γ (i.e., $\Lambda + \mathbf{k}_0 \subseteq \Gamma$) by $|\Gamma \ominus \Lambda|$:

$$\Gamma \ominus \Lambda = \{\mathbf{l} \in \Gamma \mid \mathbf{l} - \mathbf{k} \in \Gamma, \forall \mathbf{k} \in \Lambda\}. \tag{2.18}$$

This set is illustrated in Fig. 2.3 along with Γ and Λ. Since the vectors $\mathbf{c}_{\mathbf{k}-\mathbf{k}_0}$ are

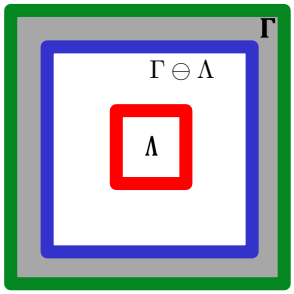


Figure 2.3. The non-minimal filter bandwidth Γ (green) is illustrated along with the minimal filter bandwidth Λ (red). The set $\Gamma \ominus \Lambda$ (blue) contains all indices at which Λ can be centered, while remaining inside Γ.

linearly independent and are orthogonal to any feature vector $\Phi_\Gamma(\mathbf{x})$ on $\mathcal{S}[\psi]$, the dimension of the subspace is bounded by

$$\dim(\mathcal{V}_\Gamma) \leq |\Gamma| - |\Gamma \ominus \Lambda|. \tag{2.19}$$

2.3.1.3 Union of irreducible surfaces with $\Gamma \supset \Lambda_i$

When $\psi = \prod_{i=1}^{m} \psi_i$, each irreducible surface $\mathcal{S}[\psi_i]$ will be mapped to a subspace of dimension $|\Gamma| - |\Gamma \ominus \Lambda_i|$. This implies that the non-linear lifting transforms the union of irreducible surfaces to the well-studied union of subspace model [44, 68, 73].

2.4 Worst-case guarantees for curve recovery

2.4.1 Annihilation relations for points on the curve

Let us now consider a set of N points on the curve, denoted by $\mathbf{x}_1, \cdots, \mathbf{x}_N$. Note that the feature maps of each one of the points satisfy the above annihilation relations, which can be compactly represented as:

$$\mathbf{c}^T \underbrace{\begin{bmatrix} \phi_\Lambda(\mathbf{x}_1) & \phi_\Lambda(\mathbf{x}_2) & \dots & \phi_\Lambda(\mathbf{x}_N) \end{bmatrix}}_{\Phi_\Lambda(\mathbf{X})} = \mathbf{0}. \tag{2.20}$$

Here, $\Phi_\Lambda(\mathbf{X})$ is the feature matrix of the points and $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_N]$.

Assume that we have a union of irreducible curves, where the bandwidth of each of the irreducible components $\mathcal{C}[\eta_i]$ is Λ_i and bandwidth of $\mathcal{C}[\psi]$ is Λ. In this case, the $|\Lambda_i|$ dimensional lifting $\Phi_{\Lambda_i}(\mathbf{x})$ of the samples on $\mathcal{C}[\eta_i]$ will lie on a $|\Lambda_i| - 1$ dimensional subspace. Similarly, the $|\Lambda|$ dimensional lifting $\Phi_\Lambda(\mathbf{x})$ of the samples on the union of irreducible curve $\mathcal{C}[\psi]$ will lie on a $|\Lambda| - 1$ dimensional subspace.

2.4.2 Curve recovery from samples

When Φ_Λ is rank-deficient by one, the coefficient vector $\mathbf{c}$ can be identified as the unique non-zero null-space basis vector of $\Phi_\Lambda(\mathbf{X})$. This implies that the features lie in an $|\Lambda| - 1$ dimensional subspace, whose normal is specified by $\mathbf{c}$. This annihilation relation is illustrated in Fig 2.2, in the context of band-limited curves considered in the next subsection.

In practice, the points are often corrupted by noise. In the presence of noise, the null-space conditions are often not satisfied exactly. In this case, we can pose the least square estimation of the coefficients from the noisy data points $\{\mathbf{x}_i\}_{i=1}^N$ as the minimization of the criterion:

$$\mathcal{C}(\mathbf{c}) = \sum_{i=1}^{N} \|\psi(\mathbf{x}_i)\|^2 = \mathbf{c}^T \mathbf{Q}_\Lambda \mathbf{c} \tag{2.21}$$

where $\mathbf{Q}_\Lambda = \sum_{i=1}^N \phi_\Lambda(\mathbf{x}_i)\phi_\Lambda(\mathbf{x}_i)^T$. To eliminate the trivial solution $\mathbf{c} = 0$, we pose the recovery as the constrained optimization scheme:

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} \mathbf{c}^T \ \mathbf{Q}_\Lambda \ \mathbf{c} \ \text{ such that } \quad \|\mathbf{c}\|^2 = 1 \tag{2.22}$$

The solution is the eigenvector corresponding to the minimum eigenvalue of $\mathbf{Q}_\Lambda$. Note that $\mathbf{Q}_\Lambda$ is nothing but $\Phi_\Lambda(\mathbf{X})\Phi_\Lambda^T(\mathbf{X})$. Thus we just need to use the singular value decomposition of $\Phi_\Lambda^T(\mathbf{X})$ to obtain the desired solution.

2.4.3 Irreducible band-limited planar curve: sampling theorem

We now focus on the problem of the recovery of the curve, given a few points $\{\mathbf{x}_i \in \mathbb{R}^2; i = 1, \cdots, N\}$ on the curve. Let us take the band-limited curve representation for the rest of the section to derive our sampling conditions. We now determine the sampling conditions for the perfect recovery of the curve $\psi(\mathbf{x}) = 0$ using (2.22). In this case, the annihilation relation is satisfied with the feature maps defined as

$$\phi_\Lambda(\mathbf{x}) \ = \ \begin{bmatrix} \exp(j\ 2\pi \mathbf{k}_1^T \mathbf{x}) \\ \vdots \\ \exp(j\ 2\pi \mathbf{k}_{|\Lambda|}^T \mathbf{x}) \end{bmatrix} \tag{2.23}$$

We also assume that Λ is a rectangular neighborhood in $\mathbb{Z}^2$ of size $k_1 \times k_2$. We first review some results from algebraic geometry.

There is a one-to-one correspondence between trigonometric polynomials and complex polynomials. We use the extension of Bézout's inequality for trigonometric polynomials, which bounds the number of solutions of the system $\mu(\mathbf{x}) = \eta(\mathbf{x}) = 0$ that do not have any common factors.

Lemma 4 (Bézout's inequality for band-limited polynomials)**.** *Let $\mu(\mathbf{x})$ and $\eta(\mathbf{x})$ be two band-limited polynomials, whose Fourier coefficients are support limited to $k_1 \times k_2$ and $l_1 \times l_2$, respectively. If μ and η have no common factor, then the system of equations*

$$\mu(\mathbf{x}) = \eta(\mathbf{x}) = 0 \tag{2.24}$$

has a maximum of $(k_1 + k_2)(l_1 + l_2) = \deg(\mu)\deg(\eta)$ solutions in $[0, 1)^2$.

The proof of Lemma 4 is given in Appendix 2.8.1. We use this property to derive our main results. We first focus on the case where ψ is an irreducible band-limited function.

Proposition 5. *Let $\{\mathbf{x}_i\}_{i=1}^N$ be N distinct points on the zero level set of an irreducible band-limited function $\psi(\mathbf{x}), \mathbf{x} \in \mathbb{R}^2$, whose Fourier coefficients are restricted to a rectangular region Λ with size $k_1 \times k_2$. Then the curve $\psi(\mathbf{x}) = 0$ can be uniquely recovered by* (2.20)*, when:*

$$N > (k_1 + k_2)^2 = \deg^2(\psi) \tag{2.25}$$

The proof is provided in Appendix 2.8.2.

Note that the sampling condition for a single irreducible curve does not specify any constraint on the distribution of points on the curve; any set of $N > (k_1 + k_2)^2$

(a) Original curve

(b) 36 random samples

(c) 36 samples in left half

(d) 36 samples in right half

Figure 2.4. Illustration of Proposition 5: We consider a curve $C[\psi]$ given by $\psi(\mathbf{x})$, where $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is support limited to a 3×3 region, shown in (a). The theorem guarantees the perfect recovery will happen if we have no less than $(k_1 + k_2)^2 = 36$ samples. We first randomly chose 36 samples on the curve. Then from these 36 randomly chosen samples, we obtained (b), which gives us perfect recovery of the original curve. Furthermore, we mentioned that we do not require any constraint on the distribution of samples on the curve. In (c), we randomly chose 36 samples from the left half part of the curve and we got perfect recovery as well. In (d), 36 samples are randomly chosen from the right half of the curve. From (d), we saw that perfect recovery of the whole curve was also obtained. For each case, the average time required for the recovery is about 1.2 second.

points are sufficient for the recovery of the curve. This proposition is illustrated in Fig. 2.4, which shows that the recovery is guaranteed irrespective of the distribution of samples. This property is similar to well-known results in non-uniform sampling of band-limited signals [72], where the recovery is guaranteed under weak conditions on the nonuniform grid and the average sampling rate exceeding Nyquist rate.

We compare this setting with the sampling conditions for the recovery of a piecewise constant image, whose gradients vanish on the zero level set of a band-limited function [85]. The minimum number of Fourier measurements required to recover the function there is $|3\Lambda|$. When $k_1 = k_2 = K$, then $9K^2$ complex Fourier samples are required, which is far more than $4K^2$ real samples required for the recovery of the curve in our setting. Note that the constant values within the regions

bounded by the curves also need to be recovered in [85], which explains the higher sampling requirement. We note that the above bounds are looser than the ones in [91], which are based on the number of available equations; they assume the chances of the equations being linear dependent is unlikely [91]. Unlike the high-probability results in [91] that our bounds are worst-case guarantees, which will hold irrespective of the sampling geometry. We note from the experiments in Fig. 2.7 that recovery succeeds in most cases whenever the number of samples exceed $|\Lambda| - 1 = k_1 k_2 - 1$, which is the number of degrees of freedom in representing the curve.

2.4.4 Union of irreducible curves: sampling theorem

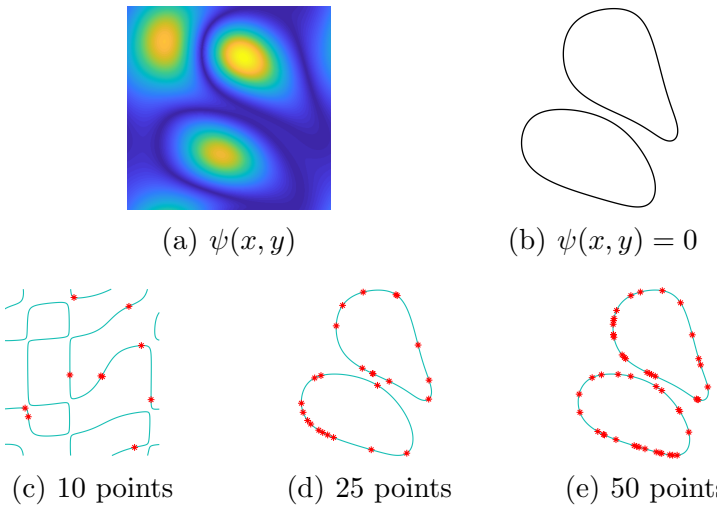
We now generalize the previous result to the setting where the composite curve is a union of multiple irreducible curves. Equivalently, the level set function is the product of multiple irreducible band-limited functions. We have the following result for this general case:

Proposition 6. *Let $\{\mathbf{x}_i\}_{i=1}^N$ be points on the zero level set of a band-limited function $\psi(\mathbf{x}), \mathbf{x} \in \mathbb{R}^2$, where the bandwidth of ψ is specified by $|\Lambda| = k_1 \times k_2$. Assume that $\psi(\mathbf{x})$ has J irreducible factors (i.e., $\psi = \eta_1 \cdots \eta_J$), where the bandwidth of the j^{th} factor is given by $k_{1,j} \times k_{2,j}$. The curve $\psi(\mathbf{x}) = 0$ can be uniquely recovered by (2.20), when each of the irreducible curves are sampled with*

$$N_j \;>\; (k_1 + k_2)(k_{1,j} + k_{2,j}) = \deg(\psi)\deg(\eta_j);\; j = 1, \cdots, J. \tag{2.26}$$

The total number of samples needed for unique recovery is specified by

$$N = \sum_{j=1}^{J} N_j = \deg(\psi) \sum_{j=1}^{J} \deg(\eta_j), \tag{2.27}$$



(a) $\psi(x, y)$ (b) $\psi(x, y) = 0$

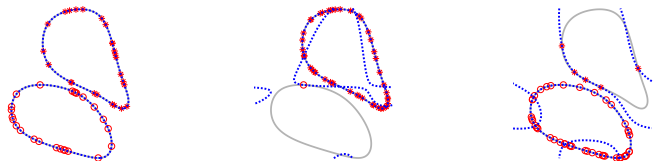
(c) 10 points (d) 25 points (e) 50 points

Figure 2.5. Illustration of Proposition 6: We consider a curve $C[\psi]$ on the top right, where $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is support limited to a 5×5 region. The level set function is shown in the top left. We consider the recovery from different number of samples of $C[\psi]$, sampled randomly. The sampling locations are marked by red crosses. Note that the theory guarantees the recovery when the number of samples exceeds $(k_1 + k_2)^2 = 100$ samples. However, we observe good recovery of the curve around 50 samples. Note that our theoretical results are worst-case guarantees, and in practice fewer samples are sufficient for good recovery as seen from Fig, 2.7. On average, the computational time required for the recovery of the curve using 50 points is about 1.5 second.

which is bounded above by $(k_1 + k_2)(k_1 + k_2 + 2(J - 1))$.

We note that the upper bound can be approximated as $(k_1 + k_2)^2$ for small values of J, which is the upper bound in Proposition 5. The above result is proved in Appendix 2.8.3. Note that unlike the case considered in Section 2.4.1, an arbitrary set of N samples cannot guarantee the perfect recovery. Each of the J irreducible curves $C[\eta_j]$ need to be sampled proportional to their complexity, specified by $\deg(\eta_j)$ to guarantee perfect recovery.

We demonstrate the above proposition in Fig. 2.5. We consider a curve $C[\psi]$, where $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is support limited to a 5×5 region. We note that there are



(a) $25 + 25$ (b) $49 + 1$ (c) $5 + 45$

Figure 2.6. Illustration of Proposition 6: We consider the same curve $C[\psi]$ as specified by Fig 2.5 (b), which is given by the union of two irreducible curves with bandwidth 3×3. So the bandwidth of $C[\psi]$ is 5×5. According to Proposition 6, we will need to have around 100 samples to recover $C[\psi]$ and each of the two irreducible curves need to satisfy with the sampling condition. As we noted in Fig 2.5, our results are worst-case guarantees. We observe that when we have 50 points and those points are uniformly sampled on the two irreducible curves, we can successfully recover the whole curve, as shown in (a). Now, if we put most of the samples on one of the irreducible curves, we cannot fully recover the curve, as illustrated in (b) and (c). This implies that the sampling condition on each of the irreducible factors is necessary in Proposition 6.

three connected components in the above curve. We consider the recovery from different number of samples of $C[\psi]$ in the middle row, sampled randomly. The random strategy ensures that the samples are distributed to the factors, roughly satisfying the conditions in Proposition 6. Note that the theory guarantees recovery, when the number of samples exceeds around $(k_1 + k_2)^2 = 100$ samples. We observe good recovery of the curve around 50 samples; note that our results are worst-case guarantees, and in practice fewer samples are sufficient for good recovery of most curves. We further study the distribution of the points in Fig. 2.6. The experiments demonstrate that each of the curves need to be sampled with a number proportional to the bandwidth of the curves as in (b). When the points are non-uniformly distributed as in (c) or (d), the recovery fails.

We further studied the above proposition in Fig. 2.7. We considered several

random curves, each with different bandwidth and considered their recovery from different number of samples. The sampling locations were picked at random. The colors indicate the average reconstruction error between the actual curve and the reconstructed curves. This reconstruction error is computed as the sum of distances between each point on one curve and the closest point to it on the other curve. We have also plotted the upper bound $(k_1 + k_2)^2$ in red, while the number of unknowns in the curve representation $k_1\, k_2$ is plotted in blue. We note that the curve can be recovered accurately when the number of samples exceed the upper bound. We also note that in general, good recovery can be obtained for most curves, when the number

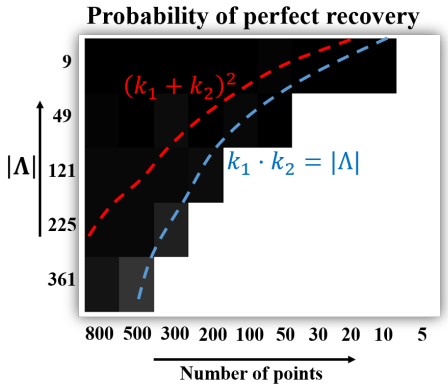


Figure 2.7. Effect of number of sampled points on perfect reconstruction. We randomly generated several curves with different bandwidth and number of sampled points, and recovered the curves from these samples. The success of reconstruction of the curves averaged over several trials are shown in the above phase transition plot, as a function of bandwidth and number of sampled entries. The color indicates the frequency of success; the color black indicates that the true curve cannot be recovered in any of the experiments, while the color white represents that the true curve is recovered in all the experiments. It is seen that perfect recovery occurs whenever we have $\geq (k_1 + k_2)^2$ samples, as indicated by our worst-case guarantees. However, we note that good recovery is observed whenever the number of samples exceed the degrees of freedom $k_1 \cdot k_2$

.

of samples exceed $k_1\, k_2$.

2.4.5 Curve recovery with unknown Fourier support

Propositions 5 and 6 assume that the true support of the Fourier coefficients of ψ, specified by Λ is known, in addition to the points $\{\mathbf{x}_i\}_{i=1}^N$. However, typically only the points will be known and the filter support will be unknown. We now consider the case where the filter support is over-estimated as $\Gamma \supset \Lambda$. We focus on the recovery of the coefficients from the annihilation relation

$$\mathbf{c}^T \mathbf{\Phi}_\Gamma \ = 0. \tag{2.28}$$

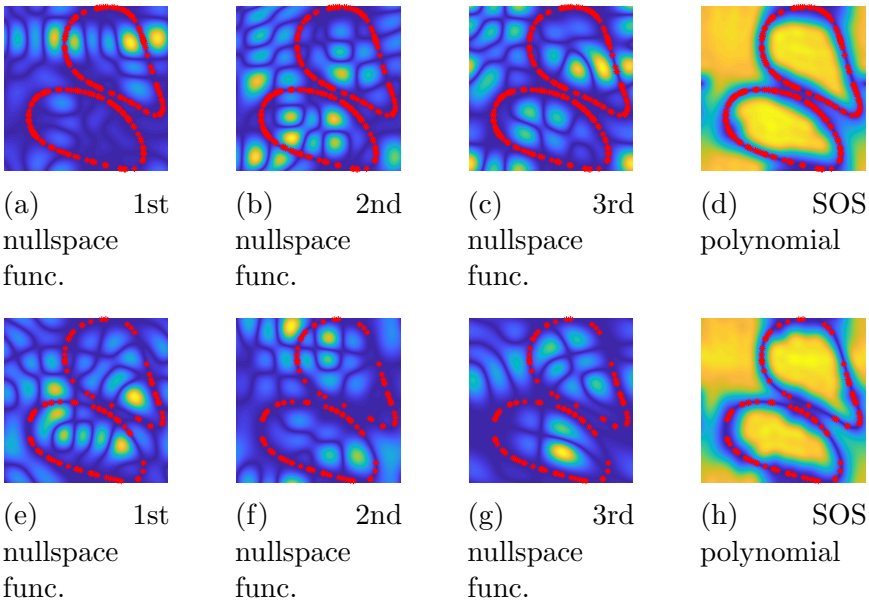
The following result shows that the above matrix will have multiple linearly independent null-space vectors. However, if the curves are sampled as described below, the corresponding band-limited functions satisfy some desirable properties that facilitate the recovery of the curves.

Proposition 7. *Consider the zero level set of the band-limited polynomial $\psi(\mathbf{x})$ with J irreducible components, as described in Proposition 6. Let the assumed bandwidth of the curve be Γ with $|\Gamma| = l_1 \times l_2$ and $\Lambda \subset \Gamma$. Then, there exist multiple functions that satisfy $\mu(\mathbf{x}_i) = 0; i = 1, \cdots, N$. If the irreducible curves of the zero level set of ψ are sampled with*

$$N_j > (l_1 + l_2)(k_{1,j} + k_{2,j}); \quad j = 1, \ldots, J, \tag{2.29}$$

all of the above functions, or equivalently the right nullspace vectors $\mathbf{c}_\mu \overset{\mathcal{F}}{\leftrightarrow} \mu$ of $\mathbf{\Phi}_\Gamma$, will be of the form:

$$\mu(\mathbf{x}) = \psi(\mathbf{x})\ \eta(\mathbf{x}) \tag{2.30}$$



(a) 1st nullspace func. (b) 2nd nullspace func. (c) 3rd nullspace func. (d) SOS polynomial

(e) 1st nullspace func. (f) 2nd nullspace func. (g) 3rd nullspace func. (h) SOS polynomial

Figure 2.8. Illustration of Propositions 7 & 8: We consider the recovery of the curve $C[\psi]$ as specified by Fig 2.5 (b), assuming unknown bandwidth. We over-estimate the support Γ as 11x11, while the original support of $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is 5×5. According to Propositions 7& 8, when the number of samples exceed $(k_1 + k_2)(l_1 + l_2) = 220$, the matrix is low-rank. The first row shows the results by using 220 samples. We display the Fourier transforms of the three null-space functions of Φ_Γ in (a), (b) and (c). This approach of visualizing the null-space functions is similar to the approaches in [47,91]. All of these functions are zero on the $\mathcal{C}[\psi]$, in addition to possessing several other zeros. The sum of squares function, denoted by (2.5.2) is shown on the right column, captures the common zeros, which specifies the curve $\mathcal{C}[\psi]$. We use the SOS function as a surrogate for the greatest common divisor of the null-space functions. Note that the bound in Proposition 7 is also a worst-case guarantee. In the second row, the curve $C[\psi]$ was sampled on 100 random sampling locations, denoted by the red crosses. We see that the curve can be recovered well using just 100 samples. The computational time used to specify the curve using SOS function in this experiment is about 1.6 second

.

where $\eta(\mathbf{x})$ *is an arbitrary function such that* $\text{supp}(\mathbf{c}_\mu) = \Gamma$.

Note that the minimal function $\psi(\mathbf{x})$ is a special case of (2.30), with $\eta = 1$. The above result is proved in Appendix 2.8.4. Since $\psi(\mathbf{x})$ is the common factor of all the annihilating functions, all of them will satisfy $\mu(\mathbf{x}) = 0$, for any point on the

original curve as well as the sampling locations. This also implies that $\psi(\mathbf{x})$ is a common divisor of the above functions $\mu(\mathbf{x})$. In fact, $\psi(\mathbf{x})$ is the greatest common divisor as we will show it in the next paragraph. We now characterize the number of linearly independent annihilation functions, or equivalently the size of the right null space of Φ_Γ.

Proposition 8. *We consider the trigonometric polynomial $\psi(\mathbf{x})$ described in Proposition 7 and $\Lambda \subset \Gamma$. Then:*

$$\operatorname{rank}\left(\Phi_\Gamma(\mathbf{X})\right) \leq \underbrace{|\Gamma| - |\Gamma : \Lambda|}_{r} \tag{2.31}$$

with equality if the sampling conditions of Proposition 7 are satisfied.

Here,

$$\Gamma : \Lambda = \{\mathbf{l} \in \Gamma : \mathbf{l} - \mathbf{k} \in \Gamma,\ \forall\ \mathbf{k} \in \Lambda\}. \tag{2.32}$$

This set is illustrated in Fig 5(a) from [85] along with Γ and Λ. The inequality of this result is same as the inequality of Proposition 5.1 in [85]. Based on the inequality, we can then obtain the second part (the equality) of the result, which provides us a means to compute the original curve, even when the original bandwidth/support Λ is unknown. Specifically, Proposition 8 shows that $\Phi_\Gamma(\mathbf{X})$ has $|\Gamma : \Lambda|$ null-space vectors, each of which satisfies (2.30). Besides, from the proof of Proposition 8, we can see that any polynomial of the form

$$\theta_{\mathbf{l}} = \exp(j2\pi\mathbf{l}^T\mathbf{x})\ \psi(\mathbf{x}), \quad \forall \mathbf{l} \in \Gamma : \Lambda \tag{2.33}$$

is a null-space vector of $\Phi_\Gamma(\mathbf{X})$. Note that the exponentials $\exp(j2\pi\mathbf{l}^T\mathbf{x}), \forall \mathbf{l} \in \Gamma : \Lambda$ are linearly independent, and hence the set $\{\theta_{\mathbf{l}}; \mathbf{l} \in \Gamma : \Lambda\}$ spans the null space of

$\Phi_\Gamma(\mathbf{X})$. Since $\exp(j2\pi\mathbf{l}^T\mathbf{x})$ does not vanish in the domain, the only common zeros of $\{\theta_\mathbf{l}; \mathbf{l} \in \Gamma : \Lambda\}$ will be the zeros of the minimal polynomial $\psi(\mathbf{x})$, meaning that $\psi(\mathbf{x})$ is the greatest common divisor of the functions that span the null-space of $\Phi_\Gamma(\mathbf{X})$. Therefore, the common zeros of these functions, or equivalently the zeros of the greatest common divisor, will specify the curve. A cheaper alternative to evaluating the greatest common divisor is to evaluate the sum of squares polynomial, specified by:

$$\gamma(\mathbf{x}) = \sum_{i=1}^{Q} \|\mu_i(\mathbf{x})\|^2 \tag{2.34}$$

which will vanish only on points satisfying $\psi(\mathbf{x}) = 0$. Here $Q = |\Gamma| - r$ is the dimension of the right null-space of Φ_Γ. Since $\mathbf{c}_\psi \overset{\mathcal{F}}{\leftrightarrow} \psi$ is a valid right null-space vector of Φ_Γ that only vanishes on the true curve, the sum of squares function γ specified in (2.5.2) will only vanish on the true curve. Thus, if the total number of points sampled are $N = \sum_{j=1}^{J} N_j > (l_1 + l_2)(k_1 + k_2 + 2(J-1))$, and are arranged as (2.29), then the curve can be uniquely recovered.

We demonstrate the above result in Fig. 2.8. We considered the sampling of the same curve illustrated in Fig. 2.5, with the exception that we over-estimated the support to be 11×11 as opposed to the true support of 5×5. We considered 220 random samples, which satisfies the sampling conditions in Proposition 7. We show three of the annihilating functions in the first three columns of Fig. 2.8. We note that all of these functions are valid annihilating functions, but possess additional zeros. By contrast, the sum of square polynomial shown on the right uniquely specifies the curve.

2.4.6 Recovery of arbitrary curves

Now, we compare our planar curves recovery results with an algorithm that relies on level set evolution. Specifically, we re-engineer the level-set based method for curve recovery termed as "distance regularized level set evolution" (DRLSE), which was introduced in [64] for image segmentation. DRLSE poses the image segmentation as the minimization of the cost function

$$\underbrace{\mathcal{E}(\phi)}_{\text{energy function}} = \lambda \underbrace{\mathcal{L}_g(\phi)}_{\text{length}} + \alpha \underbrace{\mathcal{A}_g(\phi)}_{\text{area}} + \mu \underbrace{\mathcal{R}(\phi)}_{\text{regularization}},$$

where ϕ is the level set function. Here, $\mathcal{R}_p(\phi)$ is a level-set regularization term which maintains the level-set function ϕ as a signed distance function. We choose the function to be (16) of [64]. The first and second terms are the weighted length and area of the curve, respectively:

$$\mathcal{L}_g(\phi) = \int_\Omega g\delta(\phi)|\Delta\phi| d\mathbf{x} \tag{2.35}$$

$$\mathcal{L}_g(\phi) = \int_\Omega gH(-\phi) d\mathbf{x} \tag{2.36}$$

which are determined by the choice of edge indicator function g. Length and area minimizing flows are well-studied in the level-set literature, and correspond to curve velocities that are proportional to curvature and constant velocity along the curve normals [121]. The parameter μ for level set regularization term is determined by the time step. Once the time step is chosen, μ is almost determined because of the Courant-Friedrichs-Lewy (CFL) condition. We will re-engineer DRLSE to the curve recovery from samples by choosing the edge indicator function as the distance of the level-set function to points. Since DRLSE was originally designed for image

segmentation, we use the edge-based edge indicator function discussed in (2.37). We call the re-engineered DRLSE algorithm the level-set based algorithm.

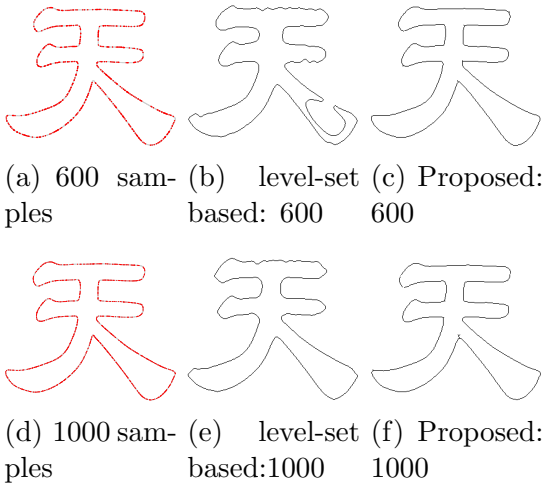
In this subsection, we compare the proposed curve recovery scheme in Section 2.5 with the level-set based algorithm. We choose the edge indicator function as the distance of the curve from the samples $\mathbf{x}_i; i = 1, .., N$:

$$g = \frac{1}{c} d(\mathbf{x}, \mathbf{x}_i) \tag{2.37}$$

where $d(\mathbf{x}, \mathbf{x}_i) = \min_{i=1,\cdots,N}\{c, \|\mathbf{x} - \mathbf{x}_i\|^2\}$ for all $\mathbf{x}$ in the image domain and c is a large constant. We compare the two methods in the context of recovering the edge curve for the Chinese character "Tian" (meaning sky in English) in Fig. 2.9.

In our method, we chose the bandwidth of the curve as 51×51. For the level-set based algorithm, we choose the parameters as $\lambda = 5$, $\alpha = 10$ and the initialization curve as a square which includes the whole curve. Note that our method do not need any initialization. The two rows show the recovery results by the two different methods from 600 and 1000 samples respectively. The first column shows the samples we choose. The results obtained by using the level-set based algorithm are given in the second column. The numbers of iterations for obtaining (b) and (e) are 1510 and 2260. The third column shows the recovery results by using our method. By comparing (b) and (c), one can see that our method recover the curve successfully from 600 randomly chosen samples. For the level-set based algorithm, the curve is not successfully recovered from those 600 samples. Once we have 1000 samples, we can find that both the two methods succeed in recovering the curve, as shown in (e) and (f). However, the computational time required for our proposed algorithm is less

than that of the level-set based algorithm. This example also demonstrates that both the two level-set based methods work well even though the curves have some sharp corners.



(a) 600 samples (b) level-set based: 600 (c) Proposed: 600

(d) 1000 samples (e) level-set based:1000 (f) Proposed: 1000

Figure 2.9. Comparison of the proposed curve recovery scheme in Section 2.4.2 with the adaptation of [64] described in Section 2.4.6. The shape is randomly sampled on the points shown in the first column. The second column consists of the curves recovered using the level-set based algorithm, while the last column shows the ones by the proposed scheme. The computational time required for the level-set based algorithm is about 66 seconds whereas the computational time required for the proposed algorithm is only about 6.4 seconds using 1000 samples.

2.4.7 Application of curve recovery in segmentation

The Mumford Shah functional is a popular formulation for segmenting objects into piecewise constant regions. It approximates an image f by a piecewise constant function

$$f = \sum_{k=1}^{K} a_k \ \chi_{\Omega_k}, \tag{2.38}$$

in the ℓ_2 sense, where $\Omega_k, k = 1, .., K$ are the regions and a_k are the constants and χ represents the characteristic function on the set. The bounded curve is denoted by $\partial\Omega = \partial\Omega_1 \cup \partial\Omega_2 \cup \cdots \cup \partial\Omega_K$. Different penalties, including the length of $\partial\Omega$ or its smoothness are imposed to regularize the optimization problem. We propose to represent $\partial\Omega$ as the zero level set of a band-limited function ψ as in [85]. In this case, the piecewise constant function satisfies $\widehat{\nabla f} * \tilde{\mathbf{c}} = 0$, which can be expressed in the matrix form as

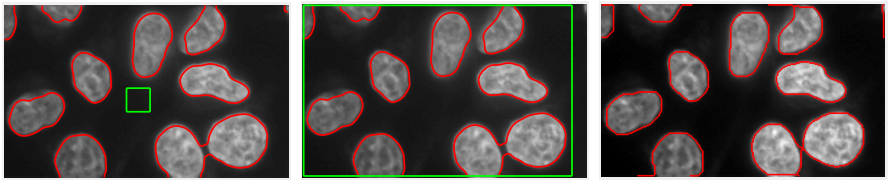
$$\mathcal{T}\left(\widehat{\nabla f}\right)\mathbf{c} = 0, \tag{2.39}$$

where $\mathcal{T}$ is a block Toeplitz 2-D convolution matrix and $\tilde{\mathbf{c}}$ is the matrix version of vector $\mathbf{c}$. When the bandwidth is over-estimated, $\mathcal{T}\left(\widehat{\nabla f}\right)$ has multiple linearly independent null-space vectors and hence the matrix is low-rank. Note that the rank of the matrix can be considered as a surrogate for the complexity of the curve $\partial\Omega$. We hence formulate the segmentation task as the low-rank optimization problem, analogous to [47].

$$f^* = \arg\min_f \|f - h\|^2 + \lambda \sum_{i=r+1}^{N} \left\|\sigma_i\left[\mathcal{T}\left(\widehat{\nabla f}\right)\right]\right\|^2 \tag{2.40}$$

where h is the original image. Note that as $\lambda \to \infty$, $\mathcal{T}\left(\widehat{\nabla f}\right)$ approaches a rank r matrix. Once f^* is obtained, the sum of square function of the null space of $\mathcal{T}\left(\widehat{\nabla f}\right)$ will specify the curve and f^* is the piecewise constant approximation. We use an alternating minimization strategy as reported in [47] to solve the above optimization scheme.

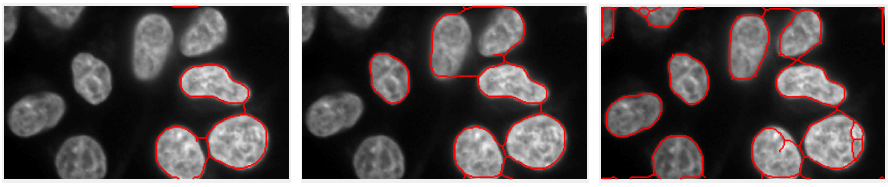
We demonstrate the preliminary utility of the segmentation scheme in Fig.



(a) DRLSE #1 (b) DRLSE #2 (c) Proposed

(d) DRLSE #1 (e) DRLSE #2 (f) Proposed

Figure 2.10. Illustration of edge based segmentation using the band-limited curve model using (2.40) and the comparisons with the segmentation method DRLSE introduced in [64]. The DLRSE scheme requires curve initialization, indicated by the green squares in the DLRSE results. The red curves in each case show the final curves. The parameters of the algorithms are optimized manually to yield the best results. The results show that the proposed scheme can provide similar segmentation as DLRSE, while it does not need initialization and is guaranteed to converge to global minimum. The ranks we choose here are 500 and 1200 for cells image and church image respectively.



(a) rank = 200 (b) rank = 350 (c) rank = 600

(d) rank = 850 (e) rank = 1000 (f) rank = 1350

Figure 2.11. Illustration of sensitivity of our proposed image segmentation algorithm to the rank. From the segmentation results, we see that when the rank is small, simpler segmentation curves will be obtained. When the rank is chosen to be too high, we will obtain over-segmentation result. Thus, the rank is a good surrogate for the complexity of the curve.

2.11 on two images: the cells image and the church image. The proposed algorithm in (2.40) is initialized with $f = h$ and iterated until convergence. The parameter λ is set to a high value (e.g. 5×10^9 in our experiments) to enforce the rank constraint. We note that the optimization scheme is capable of identifying the cells and the outline of the church, even though no curve initialization was provided. For the cells segmentation (c) and the church segmentation (f), we choose the rank to be 500 and 1200 respectively. The corresponding computational time for the two segmentations is 60 seconds and 226 seconds. We now compare our segmentation method with the level-set based segmentation method DLRSE [64], where the edge indicator function is chosen as

$$g = \frac{1}{1 + |\nabla G_\sigma * I|^2}, \tag{2.41}$$

where G_σ is a Gaussian kernel with a standard deviation σ. We considered the initialization of DLRSE with two possible curves, indicated by the green squares. The parameters in DLRSE were chosen manually to yield the best results, which corresponded to $\lambda = 6, \alpha = \pm 2, 1510$ iterations for the cells image. The parameters for the church image were $\lambda = 4.8, \alpha = -2, 2710$ iterations. For the cells image segmentation, the time required for getting (a) and (b) is 47 seconds and 40 seconds. For chruch image segmentation, the computational time for getting (d) and (e) is 175 seconds and 249 seconds. These results show that the proposed scheme is comparable to DLRSE in segmentation performance and can capture sharp features. However, the main benefit is its insensitivity to initialization, compared to DLRSE seen from (d) and (e).

2.5 High probability guarantees for surface recovery from samples

In this section, we will use the low-rank structure of the feature maps of the points to recover the surface. As discussed in the introduction, the recovery of a surface/manifold from point clouds is an important problem in denoising, machine learning, shape recovery from point clouds, and image segmentation. For presentation purposes, we consider different cases in the increasing order of complexity. In particular, we consider irreducible (single connected component) surfaces with minimal lifting, union of irreducible components with minimal lifting, and finally the case with non-minimal lifting. Note that in practice, the bandwidth of the surface is not known apriori, and hence one has to over-estimate the bandwidth; this translates to the non-minimal lifting setting. Our results in this section show that irreducible surfaces can be recovered from very few samples, as long as the number of samples exceed a number proportional to the bandwidth. Union of irreducible surfaces can also be recovered from few samples, but each of the irreducible components need to be sampled adequately to guarantee perfect recovery.

2.5.1 Sampling theorems

We consider the recovery of the surface $\mathcal{S}$ from its samples $\mathbf{x}_i; i = 1, \cdots, N$. According to the analysis in the previous section, if the sampling point $\mathbf{x}_i$ is located on the zero level set of $\psi(\mathbf{x})$, we will then have the annihilation relation specified by (2.20). Notice that equation (2.20) is a linear equation with $\mathbf{c}$ as its unknowns. Since

all the samples $\mathbf{x}_i; i = 1, .., N$ satisfy the annihilation relation (2.20), we have

$$\mathbf{c}^T \underbrace{[\Phi_\Gamma(\mathbf{x}_1) \quad \cdots \quad \Phi_\Gamma(\mathbf{x}_N)]}_{\Phi_\Gamma(\mathbf{X})} = 0. \tag{2.42}$$

We call $\Phi_\Gamma(\mathbf{X})$ the feature matrix of the sampling set $\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$. We propose to estimate the coefficients $\mathbf{c}$, and hence the surface $\mathcal{S}[\psi]$ using the above linear relation (2.42). Note that $\mathcal{S}[\psi]$ is invariant to the scale of $\mathbf{c}$; without loss of generality, we reformulate the estimation of the surface as the solution to the system of equations

$$\mathbf{c}^T \ \mathbf{\Phi}_\Gamma(\mathbf{X}) = 0; \quad \|\mathbf{c}\|_F = 1. \tag{2.43}$$

We note that without the constraint $\|\mathbf{c}\|_F = 1$, $\mathbf{c}^T \ \mathbf{\Phi}_\Gamma(\mathbf{X}) = 0$ will have a trivial solution with $\mathbf{c} = 0$. The use of the Frobenius norm constraint enables us to solve the problem using eigen decomposition. The above estimation scheme yields a unique solution, if the matrix $\Phi_\Lambda(\mathbf{X})$ has a unique null-space basis vector. We will now focus on the number of samples N and its distribution on $\mathcal{S}[\psi]$, which will guarantee the unique recovery of $\mathcal{S}[\psi]$. We will consider different lifting scenarios introduced in Section 2.3 separately. As we will see, in some cases considered below, the null-space has a large dimension. However, the minimal null-space vector (coefficients with the minimal bandwidth) will still uniquely identify the surface, provided the sampling conditions are satisfied.

2.5.1.1 Case 1: Irreducible surfaces with minimal lifting

Suppose $\psi(\mathbf{x})$ is an irreducible trigonometric polynomial with bandwidth Λ. Consider the lifting which is specified by the minimal bandwidth Λ. We see from

(2.16) that $\text{rank}\,(\Phi_\Lambda(\mathbf{X})) \leq |\Lambda| - 1$. The following result shows when the inequality is replaced by an equality.

Proposition 9. *Let $\{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ be N independent and uniformly distributed random samples on the surface $\mathcal{S}[\psi]$, where $\psi(\mathbf{x})$ is an irreducible (minimal) trigonometric polynomial with bandwidth Λ. The feature matrix $\Phi_\Lambda(\mathbf{X})$ will have rank $|\Lambda| - 1$, if*

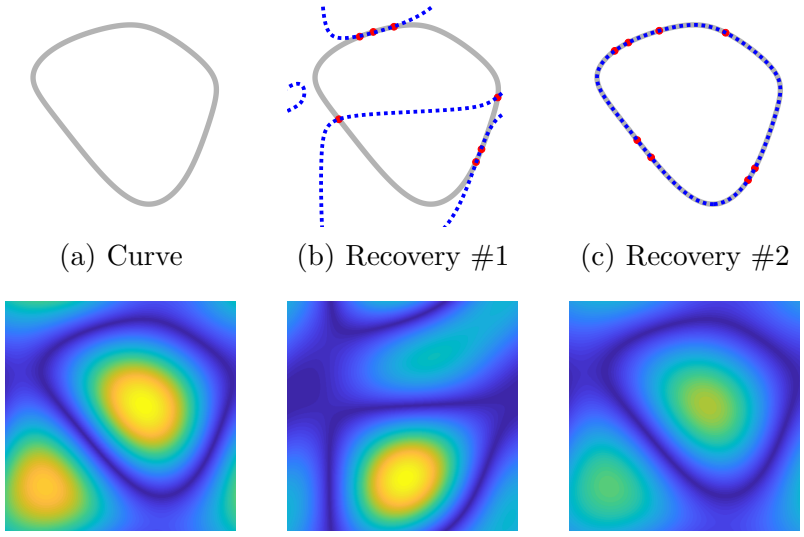
$$N \geq |\Lambda| - 1$$

for almost all surfaces $S[\psi]$.

We note that the above results are true for almost all surfaces. This implies that the surfaces for which the above results do not hold correspond to a set of measure zero [38]. The above proposition guarantees that the solution to the system of equations specified by (2.43) is unique (up to scaling) when the number of samples exceeds $N = |\Lambda| - 1$ with unit probability. The proof of this proposition can be found in Appendix 2.8.7. With Proposition 9, we obtain the following sampling theorem.

Theorem 10 (Irreducible surfaces of any dimension)**.** *Let $\psi(\mathbf{x}), \mathbf{x} \in [0,1]^n, n \geq 2$ be an irreducible trigonometric polynomial whose bandwidth is given by Λ. The zero level set of $\psi(\mathbf{x})$ is denoted as $\mathcal{S}[\psi]$. If we are randomly given $N \geq |\Lambda| - 1$ samples on $\mathcal{S}[\psi]$, then almost all surfaces $\mathcal{S}[\psi]$ can be recovered.*

This theorem generalizes the results in [155] to any dimension $n \geq 2$ and is illustrated in Fig. 2.12 and Fig. 2.13.

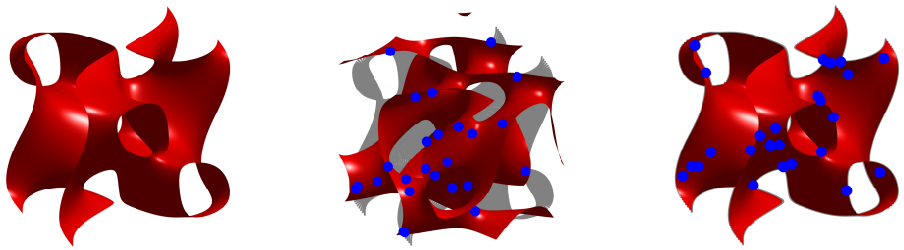


(a) Curve (b) Recovery #1 (c) Recovery #2

(d) original $\psi(x, y)$ (e) $\psi(x, y)$ with 7 samples (f) $\psi(x, y)$ with 8 samples

Figure 2.12. Illustration of Theorem 10 in 2D. The irreducible curve given by (a) is the original curve, which is obtained by the zero level set of a trigonometric polynomial whose bandwidth is 3×3. According to Theorem 10, we will need at least 8 samples to recover the curve. In (b), we randomly choose 7 samples (the red dots) on the original curve (the gray curve). The blue dashed curve shows the recovered curve from this 7 samples. Since the sampling condition is not satisfied, the recovery failed. In (c), we randomly choose 8 points (the red dots). From (c), we see that the blue dashed curve (recovered curve) overlaps the gray curve (the original curve), meaning that we recover the curve perfectly. In (d) - (f), we showed the original trigonometric polynomial, the polynomial obtained from 7 samples and the polynomial obtained from 8 samples.

In the theorem, when $n = 2$, then $\mathcal{S}$ is a planar curve. In this setting, if the bandwidth of ψ Λ is a rectangular region with dimension $k_1 \times k_2$. Then by this sampling theorem, we get perfect recovery with probability one, when the number of random samples on the curve exceeds $k_1 \cdot k_2 - 1$. Note that the degrees of freedom in the representation (2.3) is $k_1 \cdot k_2 - 1$, when we constrain $\|\mathbf{c}\|_F = 1$. This implies that if the number of samples exceed the degrees of freedom, we get perfect recovery.



(a) Surface (b) Recovery #1 (c) Recovery #2

Figure 2.13. Illustration of Theorem 10 in 3D. The irreducible surface given by (a) is the original surface, which is given by the zero level set of a trigonometric polynomial whose bandwidth is $3 \times 3 \times 3$. According to Theorem 10, we will need at least 26 samples to recover the surface. In (b), we randomly choose 25 samples (the blue dots) on the original surface (the gray part). The red surface is what we recovered from the 25 samples. Since the sampling condition is not satisfied, the recovery failed. In (c), we randomly choose 26 points (the blue dots). From (c), we see that the red surface (recovered surface) overlaps the gray surface (the original surface), meaning that we recover the surface perfectly.

Note that these results are significantly less conservative than the ones in [155], which required a minimum of $(k_1 + k_2)^2$ samples. We note that the results in [155] were the worst case guarantees, and will guarantee the recovery of the curve from any $(k_1 + k_2)^2$ samples. By contrasts, our current results are high probability results; there may exist a set of $N \geq k_1 \cdot k_2 - 1$ samples from which we cannot get unique recovery.

We note that the current work is motivated by the phase transition experiments (Fig. 5) in [155], which shows that one can recover the curve in most cases when the number of samples exceeds $k_1 \cdot k_2 - 1$ rather than the conservative bound of $(k_1 + k_2)^2$. We also note that it is not straightforward to extend the proof in [155] to the cases beyond $n = 2$. Specifically, we relied on Bezout's inequality in [155], which does not generalize easily to high dimensional cases.

2.5.1.2 Case 2: Union of irreducible surfaces with minimal lifting

We now consider the union of irreducible surfaces $\mathcal{S}[\psi]$, where ψ has several irreducible factors $\psi(\mathbf{x}) = \psi_1(\mathbf{x}) \cdots \psi_M(\mathbf{x})$. Then we have $\mathcal{S}[\psi] = \bigcup_{i=1}^{M} \mathcal{S}[\psi_i]$. Suppose the bandwidth of $\psi(\mathbf{x})$ is given by Λ and the bandwidth of each factor $\psi_i(\mathbf{x})$ is given by Λ_i. We have the following result for this setting.

Proposition 11. *Let $\psi(\mathbf{x})$ be a trigonometric polynomial with M irreducible factors, i.e.,*

$$\psi(\mathbf{x}) = \psi_1(\mathbf{x}) \cdots \psi_M(\mathbf{x}). \tag{2.44}$$

Suppose the bandwidth of each factor $\psi_i(\mathbf{x})$ is given by Λ_i and the bandwidth of ψ is Λ. Assume that $\{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ are N uniformly distributed random samples on $\mathcal{S}[\psi]$, which are chosen independently. Then with probability 1 that the feature matrix $\Phi_\Lambda(\mathbf{X})$ will be of rank $|\Lambda| - 1$ for almost all ψ if

1. *each irreducible factor is randomly sampled with $N_i \geq |\Lambda_i| - 1$ points, and*
2. *the total number of samples satisfy $N \geq |\Lambda| - 1$.*

Similar to previous propositions, the above results are valid for almost all ψ, which implies that the set of ψ for which the above results do not hold is a set of measure zero [38]. The proof of this result can be seen in Appendix 2.8.7.3. Based on this proposition, we have the following sampling conditions.

Theorem 12 (Union of irreducible surfaces of any dimension)**.** *Let $\psi(\mathbf{x})$ be a trigonometric polynomial with M irreducible factors as in* (2.45)*. If the samples $\mathbf{x}_1, ., \mathbf{x}_N$*

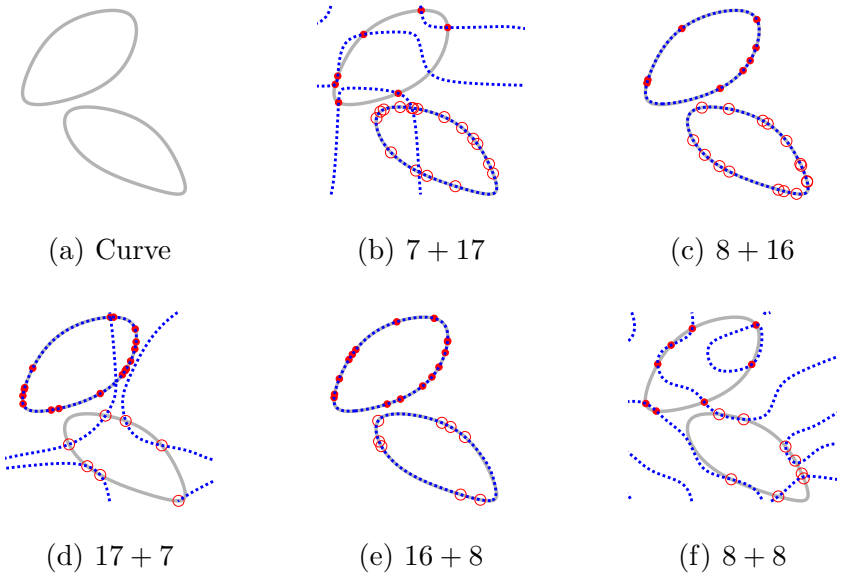
satisfy the conditions in Proposition 11, then the surface can be uniquely recovered by the solution of (2.43) *for almost all* ψ.

Unlike the sampling conditions in Theorem 10 that does not impose any constraints on the sampling, the above result requires each component to be sampled with a minimum rate specified by the degrees of freedom of that component. We illustrate the above result in Fig. 2.14 in 2D ($n = 2$), where $\mathcal{S}$ is the union of two irreducible curves with bandwidth of 3×3, respectively. The above results show that if each of these simply connected curves are sampled with at least eight points and if the total number of samples is no less than 24, we can uniquely identify the union of curves. The results show that if any of the above conditions are violated, the recovery fails; by contrast, when the number of randomly chosen points satisfy the conditions, we obtain perfect recovery.

2.5.1.3 Case 3: Non-minimal lifting

In Section 2.5.1.1 and 2.5.1.2, we introduced theoretical guarantees for the perfect recovery of the surface in any dimensions. The sampling theorems introduced in Section 2.5.1.1 and 2.5.1.2 assume that we know exactly the bandwidth of the surface or the union of surfaces. However, in practice, the true bandwidth of the surface is usually unknown. We now consider the recovery of the surface, when the bandwidth is over-estimated, or equivalently the lifting is performed assuming $\Gamma \supset \Lambda$. As discussed in Proposition 8, the dimension of $\mathcal{V}_\Gamma$ is upper bounded by $|\Gamma| - |\Gamma \ominus \Lambda|$, which implies that

$$\operatorname{rank}(\Phi_\Gamma(\mathbf{X})) \leq |\Gamma| - |\Gamma \ominus \Lambda|,$$



(a) Curve (b) $7+17$ (c) $8+16$

(d) $17+7$ (e) $16+8$ (f) $8+8$

Figure 2.14. Illustration of Theorem 12. The original curve (a) is given by the zero set of a reducible trigonometric polynomial with bandwidth 5×5, which is the product of two trigonometric polynomials with bandwidth 3×3. According to the sampling theorem, we totally need at least 24 samples and each of the components needs to be sampled for at least 8 samples. We first choose 7 samples (red dots) on the first component and 17 samples (red circles) on the second one. The gray curve in (b) is the original curve and the blue dashed curve is what we recovered from the $7+17 = 24$ samples. Since the sampling condition is not satisfied, the recovery failed. In (c), we choose 8 samples (red dots) on the first component and 16 samples (red circles) on the second one. From (c), we see that the gray curve (the original curve) overlaps the blue dashed curve (recovered curve), meaning that we recovered the curve successfully. In (d), we choose 17 samples on the first component and 7 samples on the other one. From (d), we see that the recovery is not successful. In (e), we have 16 samples on the first component and 8 samples on the second one. The original curve overlaps the recovered one. So we recovered it perfectly. Lastly, we choose 8 samples on each of the component and we failed to recover the curve as shown in (f). Note that the recovered curves pass through the samples in all cases.

where $\Gamma \ominus \Lambda$ represents the number of valid shifts of Λ within Γ as discussed in Section 2.3.1.2.

The following two propositions show when the inequality in the rank relation above can be an equality and hence we can recover the surface.

Proposition 13 (Irreducible surface with non-minimal lifting)**.** *Let* $\{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ *be* N *random samples on the surface* $\mathcal{S}[\psi]$*, chosen independently. The trigonometric polynomial* $\psi(\mathbf{x})$ *is irreducible whose true bandwidth is* Λ*. Suppose the lifting mapping is performed using bandwidth* $\Gamma \supset \Lambda$*. Then* $\operatorname{rank}(\Phi_\Gamma(\mathbf{X})) = |\Gamma| - |\Gamma \ominus \Lambda|$ *for almost all* ψ*, if*

$$N \geq |\Gamma| - |\Gamma \ominus \Lambda|.$$

The proof of this proposition can be found in Appendix 2.8.7.4.

Proposition 14 (Union of irreducible surfaces with non-minimal lifting)**.** *Let* $\psi(\mathbf{x})$ *be a randomly chosen trigonometric polynomial with* M *irreducible factors, i.e.,*

$$\psi(\mathbf{x}) = \psi_1(\mathbf{x}) \cdots \psi_M(\mathbf{x}). \tag{2.45}$$

Suppose the bandwidth of each factor $\psi_i(\mathbf{x})$ *is given by* Λ_i *and the bandwidth of* ψ *is* Λ*. Let* $\Gamma_i \supset \Lambda_i$ *be the non-minimal bandwidth of each factor* $\psi_i(\mathbf{x})$ *and* $\Gamma \supset \Lambda$ *is the bandwidth of the non-minimal lifting. Assume that* $\{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ *are* N *random samples on* $\mathcal{S}[\psi]$ *that are chosen independently. Then, the feature matrix* $\Phi_\Lambda(\mathbf{X})$ *will be of rank* $|\Gamma| - |\Gamma \ominus \Lambda|$ *for almost all* ψ *if*

1. *each irreducible factor is randomly sampled with* $N_i \geq |\Gamma_i| - |\Gamma_i \ominus \Lambda_i|$ *points, and*

2. *the total number of samples satisfy* $N \geq |\Gamma| - |\Gamma \ominus \Lambda|$.

We prove this result in Appendix 2.8.7.5. Note that in practice, when non-minimal lifting mapping is performed, we then randomly sample approximately $|\Gamma| -$

$|\Gamma \ominus \Lambda|$ positions on $\mathcal{S}$. This random strategy ensures that the samples are distributed to the factors, roughly satisfying the conditions in Proposition 14. We further studied this proposition in Fig. 2.15. We considered several random surfaces obtained by choosing random coefficients, each with different bandwidth and considered their recovery from different number of samples. From which, we obtained the phase transition plot given in Fig. 2.15, which agrees well with the theory.

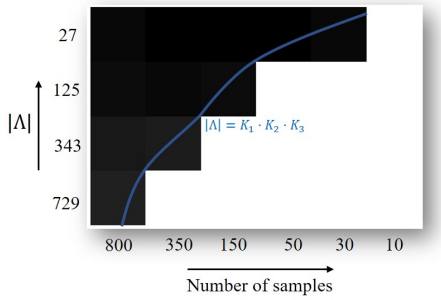


Figure 2.15. Effect of number of sampled points on surfaces reconstruction error. We randomly generated several surfaces with different bandwidths and number of sampled points, and tried to recover the surfaces from these samples. The reconstruction errors of the surfaces averaged over several trials are shown in the above phase transition plot, as a function of bandwidth and number of sampled entries. the color black indicates that the true surfaces can be recovered in any of the experiments, while the color white represents that the true surfaces are not recovered in all the experiments. It is seen that we can almost recover the surfaces with $|\Lambda| = k_1 \cdot k_2 \cdot k_3$ samples.

2.5.2 Surface recovery algorithm for the non-minimal setting

The two propositions in Section 2.5.1.3 show that $\Phi_\Gamma(\mathbf{X})$ has $|\Gamma \ominus \Lambda|$ null space basis vectors $\mathbf{n}_i \leftrightarrow \mu_i$, when the non-minimal lifting with bandwidth Γ is performed. The following result from [155] shows that the null-space vectors are related to the

minimal polynomial of the surface. In particular, all null-space vectors have the minimal polynomial as a factor. We will use this property to extract the surface from the null-space vectors as their greatest common divisor. We also introduce a simpler computational strategy which relies on the sum of squares of the null-space vectors.

Proposition 15 (Proposition 9 in [155])**.** *The coefficients of the trigonometric polynomials of the form*

$$\theta_{\mathbf{k}}(\mathbf{x}) = \exp(j2\pi\mathbf{l}^T\mathbf{x})\psi(\mathbf{x}), \quad \forall \mathbf{k} \in \Gamma \ominus \Lambda.$$

is a null space vector of $\Phi_\Gamma(\mathbf{X})$*.*

Note that the coefficients of $\theta_{\mathbf{k}}(\mathbf{x})$ correspond to the shifted versions of the coefficients of ψ and hence are linearly independent. We also note that any such function is a valid annihilating functions for points on $\mathcal{S}$. When the dimension of the null-space is $|\Gamma \ominus \Lambda|$, these corresponding coefficients form a basis for the null-space. Therefore, we have that any function in the null-space can be expressed as

$$\begin{aligned} \eta(\mathbf{x}) &= \sum_{\mathbf{k}\in\Gamma\ominus\Lambda} \alpha_{\mathbf{k}}\ \psi(\mathbf{x}) \exp(j2\pi\mathbf{k}^T\mathbf{x}) && (2.46) \\ &= \psi(\mathbf{x}) \underbrace{\sum_{\mathbf{k}\in\Gamma\ominus\Lambda} \alpha_{\mathbf{k}} \exp(j2\pi\mathbf{k}^T\mathbf{x})}_{\gamma(\mathbf{x})} = \psi(\mathbf{x})\gamma(\mathbf{x}), && (2.47) \end{aligned}$$

where $\alpha_{\mathbf{k}}$ and γ are arbitrary coefficients and function, respectively. Note that all of the functions obtained by the null-space vectors have ψ as a common factor.

Accordingly, we have that $\psi(\mathbf{x})$ is the greatest common divisor of the polynomials $\mu_i(\mathbf{x}) \leftrightarrow \mathbf{n}_i$, where $\mathbf{n}_i$ are the null-space vectors of $\Phi_\Gamma(\mathbf{X})$, which can be estimated using singular value decomposition (SVD). Since we consider polynomials

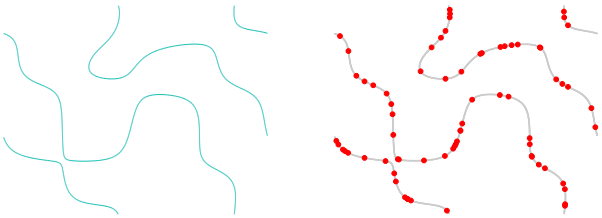
of several variables, it is not computationally efficient to find the greatest common divisor. We note that we are not interested in recovering the minimal polynomial, but are only interested in finding the common zeros of $\mu_i(\mathbf{x})$. We hence propose to recover the original surface as the zeros of the sum of squares (SoS) polynomial

$$\sigma(\mathbf{x}) = \sum_{i=1}^{|\Gamma \ominus \Lambda|} |\mu_i(\mathbf{x})|^2.$$

Note that rank guarantees in Propositions 2.8.7.4 and 2.8.7.5 ensure that the entire null-space will be fully identified by the feature matrix. Coupled with Proposition 15, we can conclude that the recovery using the above algorithm (SVD, followed by the sum of squares of the inverse Fourier transforms of the coefficients) will give perfect recovery of the surface under noiseless conditions. The algorithm is illustrated in Fig. 2.16.

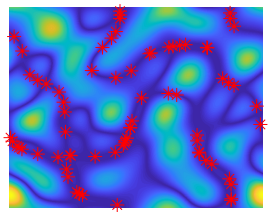
2.6 Surface recovery from noisy samples

The analysis in Section 2.5.1.3 shows that when the bandwidth of the surface is small, the feature matrix is low rank. In practice, the sampling points are usually corrupted with some noise. We denote the noisy sampling set by $\mathbf{Y} = \mathbf{X} + \mathbf{N}$, where $\mathbf{N}$ is the noise. We propose to exploit the low-rank nature of the feature matrix to recover it from noisy measurements. Specifically, when the sampling set $\mathbf{X}$ is corrupted by noise, the points will deviate from the original surface, and hence the features will cease to be low rank. We impose a nuclear norm penalty on the feature maps that will push the feature vectors to a subspace. Since the feature vectors are related to the original points by the exponential mapping, the original points will move to the surface. In practice it is difficult to compute the feature map. We hence

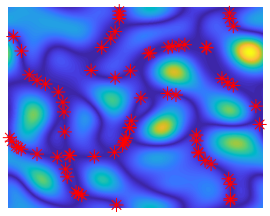


(a) The original curve

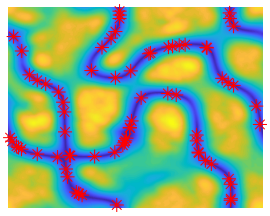
(b) The sampling points



(c) 1st null-space function



(d) 2nd null-space function



(e) sum-of-squares polynomial

Figure 2.16. Illustration of the sampling fashion for non-minimal bandwidth. We consider the curve as shown in (a), which is given by the zero level set of a trigonometric polynomial of bandwidth 5×5. We choose the non-minimal bandwidth Γ as 11×11. According to the sampling condition for non-minimal bandwidth, we sampled on 72 random locations. We randomly chose two null-space vectors for the feature matrix of the sampling set, which gave us functions (c) and (d). We can see that all of these functions have zeros on the original zero set, in addition to processing several other zeros. The sum of squares function is shown in (e), showing the common zeros, which specifies the original curve.

rely on an iterative reweighted least-squares algorithm, coupled with the *kernel-trick*, to avoid the computation of the features. Since the cost function is non-linear (due to the non-linear kernel), we use steepest descent-like algorithm to minimize the cost function. We note that each iteration of this algorithm has similarities to non-local means algorithms, which first estimate the weight/Laplacian matrix from the patches, followed by a smoothing. We also note that this approach has conceptual similarities to kernel low-rank algorithms used in MRI and computer vision [79, 81].

These algorithms rely on explicit polynomial mappings, low-rank approximation of the features, followed by the analytical evaluation of the pre-images that is possible for polynomial kernels.

We pose the denoising as:

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} ||\mathbf{X} - \mathbf{Y}||^2 + \lambda ||\Phi(\mathbf{X})||_* \tag{2.48}$$

where we use the nuclear norm of the feature matrix of the sampling set as a regularizer. Unlike traditional convex nuclear norm formulations, the above scheme is non-convex.

We adapt the kernel low-rank algorithm in [89, 99] to the high dimensional setting to solve (2.48). This algorithm relies on an iteratively reweighted least squares (IRLS) approach [42, 75] which alternates between the following two steps:

$$\mathbf{X}^{(n)} = \arg\min_{\mathbf{X}} ||\mathbf{X} - \mathbf{Y}||^2 + \lambda \text{trace}[\mathcal{K}(\mathbf{X})\mathbf{P}^{(n-1)}], \tag{2.49}$$

and

$$\mathbf{P}^{(n)} = \left[\mathcal{K}(\mathbf{X}^{(n)}) + \gamma^{(n)}\mathbf{I}\right]^{-1/2} \tag{2.50}$$

where $\gamma^{(n)} = \frac{\gamma^{(n-1)}}{\eta}$ and $\eta > 1$ is a constant. Here, $\mathcal{K}(\mathbf{X}) = \Phi_\Gamma(\mathbf{X})^T\Phi_\Gamma(\mathbf{X})$. We use the *kernel-trick* to evaluate $\mathcal{K}(\mathbf{X})$. The kernel-trick suggests that we do not need to explicitly evaluate the features. Each entry of the matrices $\mathcal{K}(\mathbf{X})$ correspond to inner-products in feature space:

$$(\mathcal{K}(\mathbf{X}))_{(i,j)} = \underbrace{\Phi(\mathbf{x}_i)^H\Phi(\mathbf{x}_j)}_{\kappa(\mathbf{x}_i,\mathbf{x}_j)} \tag{2.51}$$

which can be evaluated efficiently using the nonlinear function κ (termed as kernel function) of their inner-products in $\mathbb{R}^n$.

The dependence of the kernel function on the lifting is detailed in Section 3.2.3. Since the above problem in (2.49) is not quadratic, we propose to solve it using gradient descent as in [155]. We note that the cost function in (2.49) can be rewritten as

$$C(\mathbf{X}) = \|\mathbf{X} - \mathbf{Y}\|^2 + \lambda \sum_{i,j} \mathbf{P}_{ij}^{(n-1)} \kappa\left(\mathbf{x}_i, \mathbf{x}_j\right), \tag{2.52}$$

where $\mathbf{P}_{i,j}$ are the entries of the matrix $\mathbf{P}^{(n-1)}$. As will be discussed in detail in Section 3.2.3, the exponential kernel for a circular support as in Fig. 3.2.(b) can be approximated as a circularly symmetric kernel $\kappa(\mathbf{x}_i, \mathbf{x}_j) = k(\|\mathbf{x}_i - \mathbf{x}_j\|^2)$. In this case, the partial derivatives of (2.49) with respect to one of the vectors $\mathbf{x}_i$ is

$$\begin{aligned} \partial_{\mathbf{x}_i}\mathcal{C} &= 2(\mathbf{x}_i - \mathbf{y}_i) + 2\lambda \sum_j \underbrace{\mathbf{P}_{ij}^{(n-1)}\, k'(\|\mathbf{x}_i - \mathbf{x}_j\|^2)}_{w_{i,j}} \,(\mathbf{x}_i - \mathbf{x}_j) && (2.53) \\ &= 2(\mathbf{x}_i - \mathbf{y}_i) + 2\lambda \underbrace{\left(\sum_j w_{i,j}\right)}_{d_i} \mathbf{x}_i - \mathbf{W}\mathbf{X}. && (2.54) \end{aligned}$$

Here,

$$\mathbf{W}_{ij} = \mathbf{P}_{ij}^{(n-1)}\, k'(\|\mathbf{x}_i - \mathbf{x}_j\|^2. \tag{2.55}$$

Thus, the gradient of the cost function (2.52) is :

$$\nabla_{\mathbf{X}}\mathcal{C} \approx 2(\mathbf{X} - \mathbf{Y}) + 2\lambda \underbrace{(\mathbf{D} - \mathbf{W})}_{\mathbf{L}} \mathbf{X}. \tag{2.56}$$

Here, $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the matrix obtained from the weights $\mathbf{W}$ and $\mathbf{D}$ is a diagonal matrix with diagonal entries $d_i = \sum_j \mathbf{W}_{ij}$.

We note that the gradient of (2.49) specified by (2.56) is also the gradient of the cost function

$$\mathcal{D} = \|\mathbf{X} - \mathbf{Y}\|_F^2 + \lambda \text{ trace}\left(\mathbf{X}\ \mathbf{L}\ \mathbf{X}^H\right), \tag{2.57}$$

which is used in approaches such as non-local means (NLM) [17] and graph regularization [122]. We note that the above optimization problem is quadratic and hence has an analytical solution. We thus alternate between the solution of (2.57) and updating the weights, and hence the Laplacian matrix using (2.55), where $\mathbf{P}$ is specified by (2.50). Despite the similarity to NLM, we note that NLM approaches use a fixed Laplacian unlike the iterative approach in our work. In addition, the expression of the Laplacian is also very different. We refer the readers to [155] for comparison of the proposed scheme with the above graph regularized algorithm. Once the denoised null-space matrix is obtained from the above algorithm, we can use the sum of square approach described in Section 2.5.1.3 to recover the surfaces. We note that the algorithm is not very sensitive to the true bandwidth of the kernel Γ, as long as it over-estimates the true bandwidth of the surface Λ.

2.6.1 Point cloud denoising in 2D

We illustrate the utility of the kernel low-rank formulation (KLR) to denoise 2D points in Fig. 2.17. In Fig. 2.17, we also compared our method with another point-set denoising method called "Graph Laplacian regularized point cloud denoising (GLR)", which was introduced recently in [148]. We choose 3 examples to perform the point sets denoising algorithms and in each example, we add Gaussian noise to the point sets. In the first example, we randomly choose 409 points on the edge set of a rabbit as shown in (a). We use GLR and KLR to denoise the noisy points respectively. For GLR, we set the parameter μ to be 1000 and after 34 iterations, we obtained the denoising result (c). In KLR, we get the denoising result (d) after 80 iterations using

about 4.4 seconds. In the second examples, we choose 385 points on the edge set of a plane. We again set the parameter μ to be 1000 and after 31 iterations, we have the denoising result (g). For KLR, we get the result (h) by iterating 80 times using about 4.0 seconds. In the third example, we choose 451 points on the shape of a fish. For GLR, after 34 iterations by setting the parameter μ in the algorithm to be 1000, we obtain the denosing result (k). For our proposed denoising algorithm, we raised the number of iterations to 450 and it takes about 32.7 seconds to obtain the denoising result (l). By comparing the denosing results (c) and (d), (g) and (h), (k) and (l), we can see that both the two methods work for denosing the noisy points. While for GLR, we can see that the some points will get closer along the right curve. To compare the experimental performance mathematically, we introduce an evaluation metric, signal-to-noise ratio (SNR), for point cloud denoising. Suppose the ground-truth and predict point clouds are $\{\mathbf{x}_i\}_{i=1}^{N_1}$ and $\{\mathbf{y}_i\}_{i=1}^{N_2}$. We define the SNR, which is measured in dB by

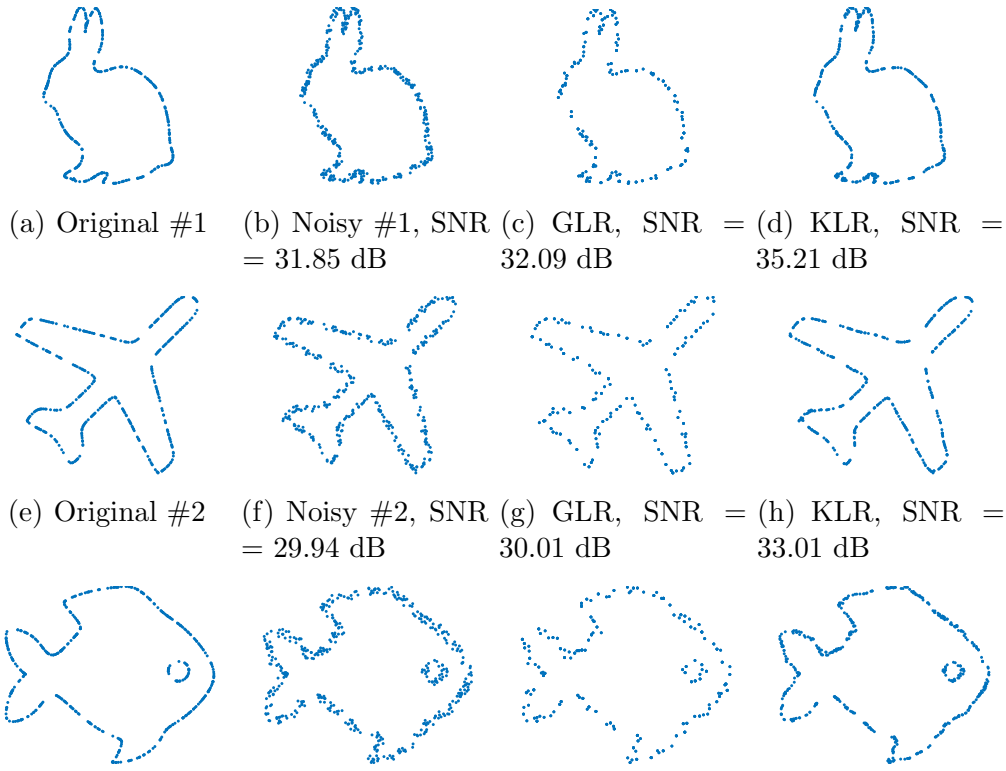
$$\text{SNR} = 10 \log \frac{1/N_2 \sum_{\mathbf{y}_i} ||\mathbf{y}_i||_2^2}{\text{MSE}},$$

where MSE is the mean-square-error defined as

$$\text{MSE} = \frac{1}{2N_1} \sum_{\mathbf{x}_i} \min_{\mathbf{y}_j} ||\mathbf{x}_i - \mathbf{y}_j||_2^2 + \frac{1}{2N_2} \sum_{\mathbf{y}_i} \min_{\mathbf{x}_j} ||\mathbf{y}_i - \mathbf{x}_j||_2^2.$$

2.6.2 Point cloud denoising in 3D

We illustrate this approach in the context of recovering 3D shapes from noisy point clouds in Fig. 2.18. The data sets are obtain from AIM@SHAPE [1]. We note that the direct approach, where the null-space vector is calculated from the noisy



(a) Original #1 (b) Noisy #1, SNR = 31.85 dB (c) GLR, SNR = 32.09 dB (d) KLR, SNR = 35.21 dB

(e) Original #2 (f) Noisy #2, SNR = 29.94 dB (g) GLR, SNR = 30.01 dB (h) KLR, SNR = 33.01 dB

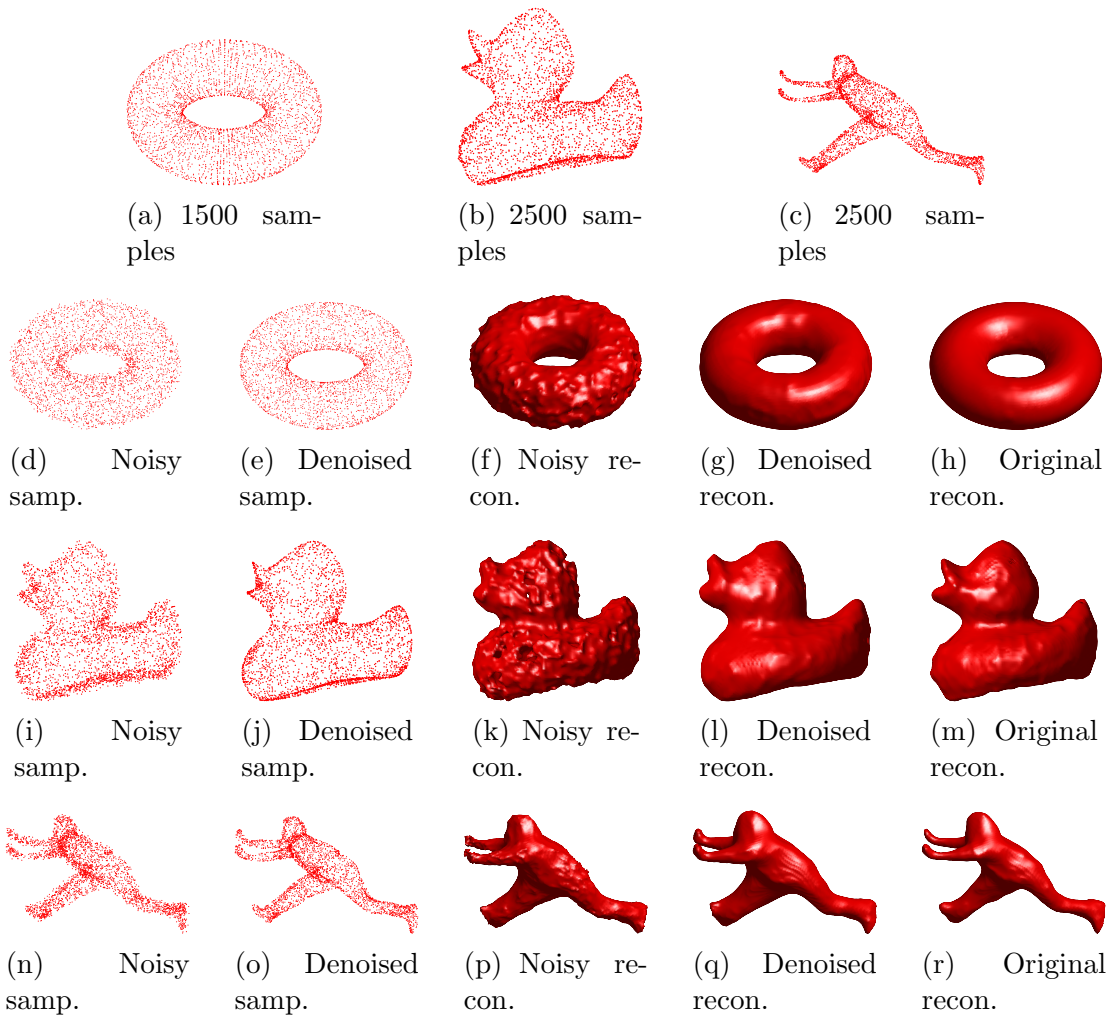
(i) Original #3 (j) Noisy #3, SNR= 28.33 dB (k) GLR, SNR = 28.95 dB (l) KLR, SNR = 31.84 dB

Figure 2.17. Comparison between proposed denoising algorithm (KLR) and Garph Laplacian Regularized denoising algorithm (GLR) introduced in [148].

feature matrix, often results in perturbed shapes. By contrast, the nuclear norm prior is able to regularize the recovery.

2.7 Discussion and Conclusion

We introduced a continuous domain framework for the recovery of points on a band-limited surfaces. The proposed bandlimited representation have several desirable geometric properties, which make it an attractive tool in a variety of shape estimation problems. We have introduced novel algorithms with sampling guarantees for the recovery of both irreducible and union of irreducible bandlimited surfaces from



(a) 1500 samples (b) 2500 samples (c) 2500 samples

(d) Noisy samp. (e) Denoised samp. (f) Noisy recon. (g) Denoised recon. (h) Original recon.

(i) Noisy samp. (j) Denoised samp. (k) Noisy recon. (l) Denoised recon. (m) Original recon.

(n) Noisy samp. (o) Denoised samp. (p) Noisy recon. (q) Denoised recon. (r) Original recon.

Figure 2.18. Illustration of the points cloud denoising algorithm and surface recovery algorithm with unknown bandwidth. The first row shows the samples drawn from three surfaces. Noise is added to the samples (see (d), (i), (n)). Then we use the proposed algorithm to denoise the points. The parameter λ in (2.48) is chosen as 1.4 for the denoising algorithm. The number of iterations for the denoising algorithm is 30. The surfaces that are recovered from noisy samples and denoised samples are also presented for comparison. The bandwidth was chosen as $31 \times 31 \times 31$ for all the experiments.

few of their samples.

We also demonstrated the utility of the representation in practical applications including image segmentation and denoising of a point cloud, which can be modeled

by a surface. The main benefit of the surface recovery from points as well as image segmentation over the state of the art is the convex formulation, which makes the algorithm insensitive to local minima errors as well as initialization. The segmentation and point cloud denoising experiments show that the proposed scheme can exploit the global structure of the points better than competing methods that rely on local surface properties such as smoothness and curvature, which makes the algorithms less sensitive to non-uniformity of sampling.

2.8 Appendix

2.8.1 Proof of Lemma 4

We first state the well-known result for complex polynomials, which we extend to the band-limited setting.

Lemma 16. *[116] Let p_1 and p_2 be two nonconstant polynomials in $\mathbb{C}[z_1, z_2]$ of degrees d_1 and d_2 respectively. If p_1 and p_2 have no common component, then the system of equations*

$$p_1 = p_2 = 0 \tag{2.58}$$

has at most $d_1 d_2$ solutions.

Lemma 4 can be proved by simply substituting $p_1 = \mathcal{P}[\mu]$ and $p_2 = \mathcal{P}[\eta]$ in Lemma 16. Specifically, the degree of $\mathcal{P}[\mu]$ and $\mathcal{P}[\eta]$ are $(k_1 + k_2)$ and $(l_1 + l_2)$ respectively. Hence, the maximum number of solutions to (2.24) is given by $(k_1 + k_2)(l_1 + l_2)$.

2.8.2 Proof of Proposition 5

Proof. The Fourier coefficients of $\psi(\mathbf{x})$ is support limited within Λ, which is the minimal support. Let $\eta(\mathbf{x})$ be another band-limited polynomial, whose Fourier coefficients are support limited within Λ and satisfies $\eta(\mathbf{x}_i) = 0$, for $i = 1, \ldots, N$. When the number of samples satisfy (2.25), this is only possible if η is a factor of ψ, according to Bézout's inequality. Thus, $\psi(\mathbf{x})$ must be a factor of $\eta(\mathbf{x})$. Since ψ is irreducible, this implies that it is the unique band-limited irreducible polynomial satisfying $\psi(\mathbf{x}_i) = 0$. □

2.8.3 Proof of Proposition 6

Proof. The polynomial $\psi(\mathbf{x})$ is represented in terms of its irreducible factors as:

$$\psi(\mathbf{x}) = \psi_1(\mathbf{x})\psi_2(\mathbf{x}) \ldots \psi_J(\mathbf{x}) \tag{2.59}$$

where the bandwidth of $\psi_j(\mathbf{x})$ is $k_{1,j} \times k_{2,j}$.

Let $\eta(\mathbf{x})$ be another polynomial with bandwidth $k_1 \times k_2$ satisfying $\eta(\mathbf{x}_i) = 0$, for $i = 1, \ldots, N$. Consider one of the irreducible sub-curves $\{\psi_j(\mathbf{x}) = 0\}$, that is sampled on N_j points satisfying (2.26). According to Lemma 4, both ψ_j and η can be simultaneously zero at these sampling locations only if ψ_j and η have a common factor. Since ψ_j is irreducible, this implies that ψ_j is a factor of η. Repeating this line of reasoning for all factors $\{\psi_j\}$, we conclude that $\psi(\mathbf{x})$ divides $\eta(\mathbf{x})$. Since both $\psi(\mathbf{x})$ and $\eta(\mathbf{x})$ have the same bandwidth, the only possibility is that $\eta(\mathbf{x})$ is a scalar multiple of $\psi(\mathbf{x})$. This implies that the curve $\psi(\mathbf{x}) = 0$ can be uniquely recovered in (2.26) is satisfied.

The total number of points to be sampled is $N = \sum_{j=1}^{J} N_j > (k_1+k_2) \sum_{j=1}^{J} (k_{1,j}+$

$k_{2,j}$).

The support of the Fourier coefficients of ψ can be expressed in terms of the supports of $\{\psi_j\}$. Using convolution properties, we get: $k_1 = 1 + \sum_{j=1}^{J}(k_{1,j} - 1)$ and $k_2 = 1 + \sum_{j=1}^{J}(k_{2,j} - 1)$. Thus, $\sum_{j=1}^{J}(k_{1,j} + k_{2,j}) = k_1 + k_2 + 2(J-1)$ and it can be concluded that $N > (k_1 + k_2)(k_1 + k_2 + 2(J-1))$. □

2.8.4 Proof of Proposition 7

Proof. Following the steps of the proof for Proposition 6, we can conclude that $\psi(\mathbf{x})$ is a factor of $\mu(\mathbf{x})$. Since $\Lambda \subset \Gamma$, it follows that $\mu(\mathbf{x}) = \psi(\mathbf{x})\ \eta(\mathbf{x})$, where $\eta(\mathbf{x})$ is some arbitrary function such that $\mu(\mathbf{x})$ is band-limited to Γ. □

2.8.5 Proof of Proposition 8

Proof. Let $\mathbf{c}$ be the minimal filter of bandwidth $|\Lambda|$, associated with the polynomial $\psi(\mathbf{x})$. We define the following filters supported in Γ for all $\mathbf{l} \in \Gamma : \Lambda$.

$$\mathbf{c_l}[\mathbf{k}] = \begin{cases} \mathbf{c}[\mathbf{k} - \mathbf{l}], & \text{if } \mathbf{k} - \mathbf{l} \in \Lambda. \\ 0, & \text{otherwise.} \end{cases} \tag{2.60}$$

$\mathbf{c_l}$ are the Fourier coefficients of $\exp(j2\pi\mathbf{l}^T\mathbf{x})\psi(\mathbf{x})$, and are all null-space vectors of the feature matrix $\Phi_\Gamma(\mathbf{X})$. The number of such filters is $|\Gamma : \Lambda|$. Hence, we get the rank bound: $\text{rank}\,(\Phi_\Gamma(\mathbf{X})) \leq |\Gamma| - |\Gamma : \Lambda|$.

If the sampling conditions of Proposition 7 are satisfied, then all the polynomials corresponding to null-space vectors of Φ_Γ are of the form: $\mu(\mathbf{x}) = \psi(\mathbf{x})\ \eta(\mathbf{x})$. Alternatively, in the Fourier domain, the filters are of the form:

$$\mathbf{c}_\mu[\mathbf{k}] = \sum_{\mathbf{l} \in \Gamma:\Lambda} \mathbf{d_l}\mathbf{c_l}[\mathbf{k}] \tag{2.61}$$

where $\mathbf{d_l}$ are the Fourier coefficients of the arbitrary polynomial $\eta(\mathbf{x})$. Thus, all the

null-space filters can be represented in terms of the basis set $\{\mathbf{c_l}\}$. This leads to the relation: $\text{rank}\,(\Phi_\Gamma(\mathbf{X})) = |\Gamma| - |\Gamma : \Lambda|$. □

2.8.6 Proof of Proposition 1

As we mentioned in Section 2.2.1.3, if we have a (hyper-)surface $\mathcal{S}$ which is given by the zero level set of a trigonometric polynomial, then there will be a minimal polynomial which defines $\mathcal{S}$ (Proposition 1). To prove this result, we need the following famous result.

Lemma 17 (Hilbert's Nullstellensatz [7])**.** *Let $\mathbb{K}$ be an algebraically closed field (for example $\mathbb{C}$). Suppose $I \subset \mathbb{K}[x_1, \cdots, x_n]$ is an ideal of polynomials, and $\mathcal{Z}(I)$ denotes the set of common zeros of all the polynomials in I. Let $\mathcal{I}(\mathcal{Z}(I))$ represents the ideal of polynomials in $\mathbb{K}[x_1, \cdots, x_n]$ vanishing on $\mathcal{Z}(I)$. Then, we have*

$$\mathcal{I}(\mathcal{Z}(I)) = \sqrt{I},$$

where $\sqrt{I}$ denotes the radical of I, specified by the set

$$\sqrt{I} =: \{p | p^n \in I, \text{ for some } n \in \mathbb{Z}^+\} \tag{2.62}$$

Remark 1. We say a set $I \subset K[x_1, \cdots, x_n]$ is an ideal, if I is closed under the addition operation (e.g. addition"+"), satisfies the associative property, has a unit element 0, and a valid inverse for every element in I. For the operation multiplication (e.g. "·"), we have $r \cdot p \in I$ and $p \cdot r \in I$ for any $r \in K[x_1, \cdots, x_n]$

Remark 2. An important property of the radical of the ideal I is that $I \subset \sqrt{I}$. Note that setting $n = 1$ in (2.62) will yield I.

Remark 3. The above lemma states that the set of all polynomials that vanish on the common zeros $\mathcal{Z}(I)$ of the polynomials in I is given by $\sqrt{I} \supset I$. Specifically, if we are given another polynomial $\eta(\mathbf{x})$ that also vanishes on the common zero set $\mathcal{Z}(I)$, then there must be positive integer n such that $\eta^n(\mathbf{x}) \in I$.

We denote the ideal generated by a function f by $(f) = \{\mu | \mu = f\gamma\}$, where γ is an arbitrary polynomial. The identity in this ideal is the zero polynomial. In particular, (f) is the family of all functions that have f as a factor. We note that the set of common zeros of all the functions in (f), denoted by $\mathcal{Z}[(f)]$ is the same as the zero set of f, denoted by $Z[f]$.

Lemma 18. *Let f, g be two polynomials in $\mathbb{C}[x_1, \cdots, x_n]$ with the same zero set. Then the two polynomials must have (up to scaling) the same factors.*

Proof. Suppose $Z[f] = Z[g] = Z$ is the zero set of f and g. Since $Z[f] = \mathcal{Z}[(f)]$, we have $\mathcal{Z}[(f)] = \mathcal{Z}[(g)] = Z$. By the Hilbert's Nullstellensatz, we have

$$\mathcal{I}(\mathcal{Z}(f)) = \sqrt{(f)}, \qquad \mathcal{I}(\mathcal{Z}(g)) = \sqrt{(g)}.$$

Since $Z(f) = Z(g)$, we then have $\mathcal{I}(\mathcal{Z}(f)) = \mathcal{I}(\mathcal{Z}(g))$ and hence $\sqrt{(f)} = \sqrt{(g)}$. As mentioned above, we have $I \subset \sqrt{I}$ for any ideal I. Therefore, we have $(f) \subset \sqrt{(f)}$ and $(g) \subset \sqrt{(g)}$. This implies that $f \in \sqrt{(f)}$ and $g \in \sqrt{(g)}$. Because we have $\sqrt{(f)} = \sqrt{(g)}$, we can obtain that $f \in \sqrt{(g)}$ and $g \in \sqrt{(f)}$. By which we have that there exist $m, n \in \mathbb{Z}$ and $p, q \in \mathbb{C}[x_1, \cdots, x_n]$ such that

$$f^n = p \cdot g, \qquad g^m = q \cdot f.$$

Therefore, we can obtain that the irreducible factors of g are of f as well and vice versa, which proves the desired conclusion. □

With this conclusion, we can now prove Proposition 1.

Proof of Proposition 1. The proof of the existence and uniqueness about ψ is same as the proof of Proposition A.3 in [88] and thus we omit them here.

In this proof, we show that $BW(\psi) \subseteq BW(\psi_1)$. Note that the algebraic surface $X = \{p = \mathcal{P}[\psi] = 0\}$ is the union of irreducible surfaces $X_j = \{p_{i_j} = 0\} \subset \mathbb{C}^n$. Define

$$\nu(x_1, \cdots, x_n) = (e^{j2\pi x_1}, \cdots, e^{j2\pi x_n}).$$

Let $\mathcal{S}_j = \nu^{-1}(X_j \cap \mathbb{T}^n)$. Then we have a decomposition of $\mathcal{S}$ as the union of surfaces $\mathcal{S}_j$. If ψ_1 is another trigonometric polynomial with $\mathcal{S}$ as the zero level set as well. Then ψ_1 vanishes on each $\mathcal{S}_j$. Let $q = \mathcal{P}[\psi_1]$. Then we have $q = 0$ on the infinite set $\nu(\mathcal{S}_j)$, by which we can infer that q and p will have the same zero set using Theorem 19. Then by Lemma 18, we have $p \mid q$, which implies that $BW(\psi) \subseteq BW(\psi_1)$. □

2.8.7 Proof of results in Section 2.5

The key property of surfaces that we exploit is that the dimension of the intersection of two band-limited surfaces of dimension k is strictly lower than k, provided their level set functions do not have any common factors. Hence, if we randomly sample one of the surfaces, the probability that the samples fall on the intersection of the two surfaces is zero. This result enables us to come up with the sampling guarantees. We will now show the results about the intersections of the zero sets of two trigonometric surfaces.

2.8.7.1 Intersection of surfaces

We will first state a known result about the intersection of the zero sets of two polynomials (non-trigonometric) whose level set functions do not have a common factor.

Theorem 19 ([46],pp.115, Theorem 14)**.** *Let $\mathcal{S}[\psi]$ and $\mathcal{S}[\eta]$ be two surfaces of dimension $n-1$ over a field $\mathbb{K}$, which are the zero sets of the polynomials $\psi : \mathbb{K}^n \to \mathbb{K}$ and $\eta : \mathbb{K}^n \to \mathbb{K}$, respectively. If ψ and η do not have a common factor, then*

$$\dim\left(\mathcal{S}[\psi] \cap \mathcal{S}[\eta]\right) < n-1.$$

The above result is a generalization of the two dimensional case ($\mathbb{C}^2$) in [88], where Bézout's inequality was used to prove the result. Specifically, the result in [88] suggests that the intersection of two curves consists of a set of isolated points, if their potential function does not have any common factor. Theorem 19 generalizes the above result to $n > 2$; it suggests that the intersection of two surfaces with dimension k is another surface, whose dimension is strictly less than k. For instance, the intersection of two 3-D surfaces which are given by the zero level set of some polynomials, could yield 2D curves or isolated points. We now extend Theorem 19 to trigonometric polynomials using the mapping ν specified by (2.4).

Lemma 20. *Let $\mathcal{S}[\psi]$ and $\mathcal{S}[\eta]$ within $[0,1]^n \subset \mathbb{R}^n$ be two surfaces of dimension $n-1$ over $\mathbb{R}$, which are the zero level sets of the trigonometric polynomials ψ and η. Suppose ψ and η do not have a common factor, then*

$$\dim(\mathcal{S}[\psi] \cap \mathcal{S}[\eta]) < n-1.$$

Proof. Let $\nu = (\nu_1, \cdots, \nu_n)$ be defined by (2.4). We now would like to prove the result by way of contradiction. Suppose

$$\dim(\mathcal{S}[\psi] \cap \mathcal{S}[\eta]) = \dim(\mathcal{S}[\psi]) = \dim(\mathcal{S}[\eta]) = n - 1.$$

This implies that $\nu(\mathcal{S}[\psi] \cap \mathcal{S}[\eta])$ will have the same dimension of $\nu(\mathcal{S}[\psi])$ and $\nu(\mathcal{S}[\eta])$. However, this is impossible according to Theorem 19. Therefore, we have the desired result. □

Based on this lemma, we can directly have the following Corollary.

Corollary 21. *Suppose $\psi(\mathbf{x}), \eta(\mathbf{x}), \mathbf{x} \in [0,1]^n$ are two trigonometric polynomials as in Lemma 20. Consider the $n-1$ dimensional Lebesgue measure on $\mathcal{S}[\psi]$. Then this Lebesgue measure of the intersection of the zero level sets of the trigonometric polynomials is zero, i.e.,*

$$m(\mathcal{S}[\psi] \cap \mathcal{S}[\eta]) = 0.$$

The Lebesgue measure can be viewed as the area of the $n-1$ dimensional surface. For example, when $n = 3$, $\mathcal{S}[\psi]$ and $\mathcal{S}[\eta]$ are 2-D surfaces, while their intersection is a 1-D curve or a set of isolated points with zero area.

2.8.7.2 Proof of Proposition 9

Proof. We note that $N \geq |\Lambda| - 1$ is a necessary condition for the matrix to have a rank of $|\Lambda| - 1$. We now assume that the surface is sampled with $N \geq |\Lambda| - 1$ random samples, chosen independently, denoted by $\mathbf{x}_i; i = 1, \cdots, N \in \mathcal{S}[\psi]$. Since $\mathbf{c} \leftrightarrow \psi$ is a valid non-trivial null-space vector for the feature matrix $\Phi_\Lambda(\mathbf{X})$ formed from these

samples, we have rank $(\Phi_\Lambda(\mathbf{X})) \leq |\Lambda| - 1$. The polynomial $\psi(\mathbf{x}) = \mathbf{c}^T\Phi_\Lambda(\mathbf{x})$ is the minimal irreducible polynomial that defines the surface.

We now prove the desired result by contradiction. Assume that these exists another linearly independent null-space vector $\mathbf{d} \leftrightarrow \eta$ or equivalently the rank of $\Phi_\Lambda(\mathbf{X})$ is strictly less than $|\Lambda| - 1$. Since $\mathbf{c}$ and $\mathbf{d}$ are linearly independent and $\psi(\mathbf{x})$ is the minimal polynomial, we know that $\psi(\mathbf{x})$ and $\eta(\mathbf{x})$ will not share a common factor. Also note that $\mathbf{x}_i \in \mathcal{S}[\psi] \cap \mathcal{S}[\eta]$. However, since $\psi(\mathbf{x})$ and $\eta(\mathbf{x})$ do not share a common factor, the probability of each sample to be at the intersection of the two polynomials ($\mathbf{x}_i \in \mathcal{S}[\psi] \cap \mathcal{S}[\eta]$) is zero by Corollary 21. Therefore, with probability 1 that such $\mathbf{d}$ does not exist, meaning that with probability 1 that the feature matrix will be of rank $|\Lambda| - 1$ when $N \geq |\Lambda| - 1$. □

2.8.7.3 Proof of Proposition 11

Proof. We note that $N \geq |\Lambda| - 1$ is a necessary condition for the matrix to have a rank of $|\Lambda| - 1$. We now assume that the surface is sampled with N random samples $\mathbf{x}_i; i = 1, \cdots, N$ satisfying the conditions in Proposition 11. The minimal polynomial $\psi(\mathbf{x}) = \mathbf{c}^T\Phi_\Lambda(\mathbf{x})$ that defines the surface can be factorized as $\psi(\mathbf{x}) = \psi_1(\mathbf{x}) \cdot \psi_2(\mathbf{x}) \cdots \psi_M(\mathbf{x})$.

We will prove the result by contradiction. Assume that these exists another linearly independent null-space vector $\mathbf{d} \leftrightarrow \eta$, or equivalently the rank of $\Phi_\Lambda(\mathbf{X})$ is less than $|\Lambda| - 1$. Since $\mathbf{c}$ and $\mathbf{d}$ are linearly independent, ψ and η should differ by at least one factor. Without loss of generality, let us assume that $\eta(\mathbf{x}) = \mu(\mathbf{x}) \prod_{i=1}^{M-1} \psi_i(\mathbf{x})$, where μ is an arbitrary polynomial of bandwidth Λ_M. Besides, μ and ψ_M does not

share a factor. Using the result of Proposition 9, we see that the probability of μ and an irreducible ψ_M vanish at $|\Lambda_i| - 1$ independently drawn random locations is zero. If multiple factors are shared, the same argument can be extended to each one of the factors independently. □

2.8.7.4 Proof of Proposition 13

Proof. We note that $N \geq |\Gamma| - |\Gamma \ominus \Lambda|$ is a necessary condition for the matrix to have the specified rank. We now assume that the surface is sampled with $N \geq |\Gamma| - |\Gamma \ominus \Lambda|$ random samples, chosen independently. We note that $\mathbf{c} \leftrightarrow \psi$ specified by (2.7), as well as the $|\Gamma \ominus \Lambda|$ translates of $\mathbf{c}$ within Γ, are valid linearly independent null-space vectors of $\Phi_\Lambda(\mathbf{X})$. We thus have

$$\text{rank}\left(\Phi_\Lambda(\mathbf{X})\right) \leq |\Gamma| - |\Gamma \ominus \Lambda| \tag{2.63}$$

We will show that the rank condition can be satisfied with probability 1 by contradiction. Assume that these exists another linearly independent null-space vector $\mathbf{d} \leftrightarrow \eta$ or equivalently the rank of $\Phi_\Lambda(\mathbf{X})$ is less than $|\Gamma| - |\Gamma \ominus \Lambda|$. Since $\mathbf{d}$ are linearly independent with $\mathbf{c}$ and its translates within Γ, we cannot express $\mathbf{d}$ as the linear combinations of the the other null-space vectors. Specifically, we have

$$\eta(\mathbf{x}) \neq \sum_{\mathbf{k} \in \Gamma \ominus \Lambda} \alpha_{\mathbf{k}}\ \psi(\mathbf{x}) \exp(j2\pi \mathbf{k}^T \mathbf{x}) \tag{2.64}$$

$$= \psi(\mathbf{x}) \underbrace{\sum_{\mathbf{k} \in \Gamma \ominus \Lambda} \alpha_{\mathbf{k}} \exp(j2\pi \mathbf{k}^T \mathbf{x})}_{\gamma(\mathbf{x})} = \psi(\mathbf{x})\gamma(\mathbf{x}). \tag{2.65}$$

Here $\alpha_{\mathbf{k}}$ is an arbitrary coefficients and hence γ is an arbitrary polynomial. The linear independence property implies that $\eta(\mathbf{x})$ cannot have $\psi(\mathbf{x})$ as a factor. Since $\psi(\mathbf{x})$

is the minimal polynomial, this also means that η and ψ does not have any common factor.

Consider now the random sampling set $\mathbf{x}_i; i = 1..|\Gamma| - |\Gamma \ominus \Lambda|$. We have

$$\mathbf{c}^T \Phi_\Lambda(\mathbf{x}_i) = \mathbf{d}^T \Phi_\Lambda(\mathbf{x}_i) = 0, \;\; i = 1, \cdots, |\Lambda| - 1.$$

This implies that $\mathbf{x}_i \in \mathcal{S}[\psi] \cap \mathcal{S}[\eta]$. However, since $\psi(\mathbf{x})$ and $\eta(\mathbf{x})$ do not share a common factor, the probability of each sample to be at the intersection of the two polynomials ($\mathbf{x}_i \in \mathcal{S}[\psi] \cap \mathcal{S}[\eta]$) is zero by Corollary 21. Therefore, we have $\text{rank}\,(\Phi_\Lambda(\mathbf{X})) = |\Gamma| - |\Gamma \ominus \Lambda|$ with probability one.

□

2.8.7.5 Proof of Proposition 14

Proof. We note that $N \geq |\Gamma| - |\Gamma \ominus \Lambda|$ is a necessary condition for the matrix to have the specified rank. We now assume that the surface is sampled with N random samples satisfying the sampling conditions in Proposition 14. The minimal polynomial $\psi(\mathbf{x}) = \mathbf{c}^T \Phi_\Lambda(\mathbf{x})$ that defines the surface can be factorized as $\psi(\mathbf{x}) = \psi_1(\mathbf{x}) \cdot \psi_2(\mathbf{x}) \cdots \psi_M(\mathbf{x})$.

Assume that there exists another linearly independent null-space vector $\mathbf{d} \leftrightarrow \eta$ or equivalently the rank of $\Phi_\Lambda(\mathbf{X})$ is less than $|\Gamma| - |\Gamma \ominus \Lambda|$. Similar to the above arguments, if η and ψ does not have any common factors, the rank condition is satisfied with probability 1. Similar to Section 2.8.7.3, linear independence implies that $\eta(\mathbf{x})$ cannot be a factor of ψ; there is at least one factor ψ_i that is distinct. Based on Proposition 13, these factors cannot vanish on more than $|\Gamma_i| - |\Gamma_i \ominus \Lambda_i|$ common samples. □

CHAPTER 3

LEARNING FUNCTIONS ON UNION OF SURFACES: LINKS TO NEURAL NETWORK

3.1 Introduction

Several imaging algorithms were introduced to exploit the extensive redundancy with images to recover them from noisy and possibly undersampled measurements. For instance, several patch-based image denoising methods were introduced in the recent past. Algorithms such as non-local means perform averaging of similar patches within the image to achieve denoising [17]. Similar patch-based regularization strategies are used for image recovery from undersampled data [77, 143]. Similar approaches are also used for the recovery of images in a time series by exploiting their non-local similarity [97, 99]. The success of these methods could be attributed to the manifold assumption [39, 117], which states that signals in real-world datasets (e.g. patches in images) are restricted to smooth manifolds in high dimensional spaces. In particular, the regularization penalty used in non-local methods can be viewed as the energy of the signal gradients on the patch manifold rather than in the original domain, facilitating the collective recovery of the patch manifold from noisy measurements [11]. In particular, non-local methods estimate the interpatch weights, which are used for denoising; the interpatch weights are equivalent to the manifold Laplacian, which captures the structure of the manifold. Similarly, image denoising approaches such as BM3D [26] that cluster patches, followed by PCA approximations of the cluster, can also be viewed as modeling the tangent subspaces of the patch

manifold in each neighborhood. Patch dictionary based schemes, which allow the coefficients to be adapted to the specific patch, could also be viewed as tangent subspace approximation methods.

Convolutional neural networks are now emerging as very powerful alternatives for image denoising [126, 150] and image recovery [3, 50]. Rather than averaging similar patches, neural networks learn how to denoise the image neighborhoods from example pairs of noisy and noise-free patches. These frameworks can be viewed as learning a multidimensional function in high dimensional patch spaces. In particular, the inputs to the network are noisy patches and the corresponding outputs are the denoised patches/pixels. We note that the learning of such functions using conventional methods will suffer from the curse of dimensionality. Specifically, large amounts of training data may be needed to learn the parameters of such a high-dimensional function, if represented using conventional methods. While the empirical performance of neural networks is impressive, the mathematical understanding of why and how they can learn complex multidimensional functions in high-dimensional spaces from relatively limited training data is still emerging. We note that the manifold assumption is also used in the CNN literature to explain the good performance of neural networks.

With the goal of understanding the above algorithms from a geometrical perspective, we consider the following conceptual problems **(a)** when can we exactly learn and recover a function that lives on a surface, from few input-output examples, **(b)** can these results explain the good performance of imaging algorithms that use manifold structure. We note that many different surface models including parametric

shape models [52,65], local and multi-resolution representations [93,110], and implicit level-set [64,90,109] shape representations have been used in low-dimensional settings (e.g. 2D/3D).

In the previous chapter, we show that under the above assumptions on the surface, a non-linear mapping of the points on the surface will live on a low-dimensional feature subspace, whose dimension depends on the complexity of the surface. Specifically, one can transform each data point to a feature vector, whose size is equal to the number of basis functions used for the surface representation. Since we use a linear combination of complex exponentials to represent the surface, the lifting in our setting is an exponential mapping. We use the low-rank property of the feature matrix to estimate the surface from few of its samples. Our sampling results show that an irreducible surface can be perfectly recovered from very few samples, whose number is dependent on the bandwidth.

We now show that the low-rank property can be used to efficiently represent multidimensional functions of points living on the surface. In particular, we are only interested in the good representation of the function when the input is on or in the vicinity of the surface. We assume the functions are linear combination of the same basis functions (exponentials in our case). Since such representations are linear in the feature space, the low-rank nature of the exponential features provides an elegant approach to represent the function using considerably fewer parameters. In particular, we show that the feature vectors of a few anchor points on the surface span the space, which allows us to efficiently represent the function as the interpolation of the function

values at the anchor points using a Dirichlet kernel. The significant reduction in the number of free parameters offered by this local representation makes the learning of the function from finite samples tractable. We note that the computational structure of the representation is essentially a one-layer kernel network. Note that the approximation is highly local; the true function and the local representation match only on the surface, while they may deviate significantly on points which are not on the surface. We demonstrate the preliminary utility of this network in denoising, which shows improved performance compared to some state-of-the-art methods. Here, we model the denoiser as a function $f : \mathbb{R}^{p^2} \to \mathbb{R}$ that provides a *noise-free* center pixel of a $p \times p$ noisy patch. The noisy patch is assumed to a point in p^2 dimensional space, close to the low-dimensional patch surface or union of surfaces. We also show that this framework can be used to learn a manifold, which can be viewed as the signal subspace version of the null-space based kernel low-rank algorithm considered above. In this case, the network structure is an auto-encoder.

This work is related to kernel methods, which are widely used for the approximation of functions [11, 23, 70]. It is well-known that an arbitrary function can be approximated using kernel methods, and the computational structure resembles a single hidden layer neural network. Our work has two key distinctions with the above approaches: (a) unlike most kernel methods that choose infinite bandwidth kernels (e.g. Gaussians), we restrict our attention to a band-limited kernel. (b) We focus on a restrictive data model, where the data samples are localized or close to the zero set of a band-limited function. We focus on bandlimited surfaces in this work to borrow

the theoretical tools from the previous chapter. We stress that our main focus is on high-dimensional ($\gg 3$) extensions of the level set approach and generalization to shape recovery. Non-parametric and even parametric level-set methods [13, 140] will be associated with very high computational complexity in this setting without the proposed computational approaches, and has not been reported to the best of our knowledge.

3.2 Recovery of functions on surfaces

As discussed in the introduction, modern machine learning algorithms pre-learn functions from given input and output data pairs [59]. For example, CNN based denoising approaches that provide state-of-the-art results essentially learn to generate noise-free pixels or patches from given training data with several noisy and noise-free patch pairs [126, 150]. The problem can be formulated as estimating a nonlinear function $\mathbf{y} = f(\mathbf{x})$, given input and output data pairs $(\mathbf{x}_i, \mathbf{y}_i); i = 1, .., N_{\text{train}}$. A challenge in the representation of such high dimensional function is the large number of parameters, which is also termed as the curse of dimensions. Kernel methods [94], random forests [115] and neural networks [149] provide a powerful class of machine learning models that can be used in learning highly nonlinear functions. These models have been widely used in many machine learning tasks [49].

We now show that the results shown in the previous chapter provide an attractive option to compactly represent functions, when the data lie on a smooth surface or manifold in high dimensional spaces. We note that the manifold assumption is widely assumed in a range of machine learning problems [39, 43]. We now show that

if the data lie on a smooth surface in high dimensional space, one can represent the multidimensional functions very efficiently using few parameters.

We model the function using the same basis functions used to represent the level set function. In our case[1], we model it as a band-limited multidimensional function:

$$f(\mathbf{x}) = \sum_{\mathbf{k}\in\Gamma} \beta_{\mathbf{k}} \exp(j2\pi\mathbf{k}^T\mathbf{x}) = \boldsymbol{\beta}^T \Phi_\Gamma(\mathbf{x}), \tag{3.1}$$

where $\mathbf{x} \in \mathbb{R}^n$. The number of free parameters in the above representation is $|\Gamma|$, where $\Gamma \subset \mathbb{Z}^n$ is the bandwidth of the function. Note that $|\Gamma|$ grows rapidly with the dimension n. The large number of parameters needed for such a representation makes it difficult to learn such functions from few labeled data points. We now show that if the points lie on the union of irreducible surfaces as in (2.11), where the bandwidth of ψ is given by $\Lambda \subset \Gamma$, we can represent functions of the form (3.1) efficiently.

3.2.1 Compact representation of features using anchor points

We use the upper bound of the dimension of the feature matrix in (2.19) to come up with an efficient representation of functions of the form 3.1. The dimension bound (2.19) implies that the features of points on $S[\psi]$ lie in a subspace of dimension $r = |\Gamma| - |\Gamma \ominus \Lambda|$, which is far smaller than $|\Gamma|$ especially when the dimension n is large. We note that kernel methods often approximate the feature space using few eigen vectors of kernel PCA. However, there is no guarantee that these basis vectors are mappings of some points on $\mathcal{S}$. Hence, it is a common practice to consider all the

[1]We note that similar results can be obtained when the function f and the level set function are represented as a linear combination of shift-invariant functions or polynomials.

training samples to capture the low-dimensional feature vectors in kernel PCA. We now show that it is possible to find a set of $N \geq r$ anchor points $\mathbf{a}_1, \cdots, \mathbf{a}_N \in \mathcal{S}[\psi]$, such that the feature space $\mathcal{V}_\Gamma(\mathcal{S})$ is in $\text{span}\{\Phi_\Gamma(\mathbf{a}_1), \cdots, \Phi_\Gamma(\mathbf{a}_N)\}$. This result is a Corollary of Proposition 14.

Corollary 22. *Let $\psi(\mathbf{x})$ be a randomly chosen trigonometric polynomial with M irreducible factors as in (2.45). Suppose $\Gamma_i \supset \Lambda_i$ is the non-minimal bandwidth of each factor $\psi_i(\mathbf{x})$ and $\Gamma \supset \Lambda$ is the total bandwidth. Let $\{\mathbf{a}_1, \cdots, \mathbf{a}_N\}$ be N randomly chosen anchor points on $\mathcal{S}[\psi]$ satisfying*

1. *each irreducible factor $\mathcal{S}[\psi_i]$ is sampled with $N_i \geq |\Gamma_i| - |\Gamma_i \ominus \Lambda_i|$ points, and*
2. *the total number of samples satisfy $N \geq |\Gamma| - |\Gamma \ominus \Lambda|$.*

Then,

$$\mathcal{V}_\Gamma(\mathcal{S}) \subseteq \text{span}\left\{\Phi_\Gamma(\mathbf{a}_i); i = 1, \cdots, N\right\} \tag{3.2}$$

with probability 1.

As discussed in Section 2.5.1.3, if we randomly choose $N \geq |\Gamma| - |\Gamma \ominus \Lambda| = r$ points on $\mathcal{S}[\psi]$, the feature matrix will satisfy the conditions in Corollary 22 and hence (3.2) with unit probability. This relation implies that the feature vector of any point $\mathbf{x} \in \mathcal{S}[\psi]$ can be expressed as the linear combination of the features of the

anchor points $\Phi_\Gamma(\mathbf{a}_i); i = 1, \cdots, N$:

$$\Phi_\Gamma(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i(\mathbf{x})\ \Phi_\Gamma(\mathbf{a}_i) \tag{3.3}$$

$$= \underbrace{\begin{bmatrix}\Phi_\Gamma(\mathbf{a}_1) & \cdots & \Phi_\Gamma(\mathbf{a}_N)\end{bmatrix}}_{\Phi(\mathbf{A})} \underbrace{\begin{bmatrix}\alpha_1(\mathbf{x}) \\ \vdots \\ \alpha_N(\mathbf{x})\end{bmatrix}}_{\boldsymbol{\alpha}(\mathbf{x})} \tag{3.4}$$

Here, $\alpha_i(\mathbf{x})$ are the coefficients of the representation. Note that the complexity of the above representation is dependent on N, which is much smaller than $|\Gamma|$, when the surface is highly band-limited. We note that the above compact representation is exact only for $\mathbf{x} \in \mathcal{S}[\psi]$ and not for arbitrary $\mathbf{x} \in \mathbb{R}^n$; the representation in (3.4) will be invalid for $\mathbf{x} \notin \mathcal{S}[\psi]$.

However, this direct approach requires the computation of the high dimensional feature matrix, and hence may not be computationally feasible for high dimensional problems. We hence consider the normal equations and solve for $\boldsymbol{\alpha}(\mathbf{x})$ as

$$\boldsymbol{\alpha}(\mathbf{x}) = \left(\underbrace{\Phi(\mathbf{A})^H \Phi(\mathbf{A})}_{\mathcal{K}(\mathbf{A})}\right)^\dagger \underbrace{\left(\Phi(\mathbf{A})^H \Phi_\Gamma(\mathbf{x})\right)}_{\mathbf{k}_\mathbf{A}(\mathbf{x})}, \tag{3.5}$$

where $(\cdot)^\dagger$ denotes the pseudo-inverse.

3.2.2 Representation and learning of functions

Using (3.1), (3.4), and (3.5), the function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ can be written as

$$\mathbf{f}(\mathbf{x}) = \boldsymbol{\beta}^T\ \Phi(\mathbf{A})\ \mathcal{K}(\mathbf{A})^\dagger\ \mathbf{k}_\mathbf{A}(\mathbf{x}) \tag{3.6}$$

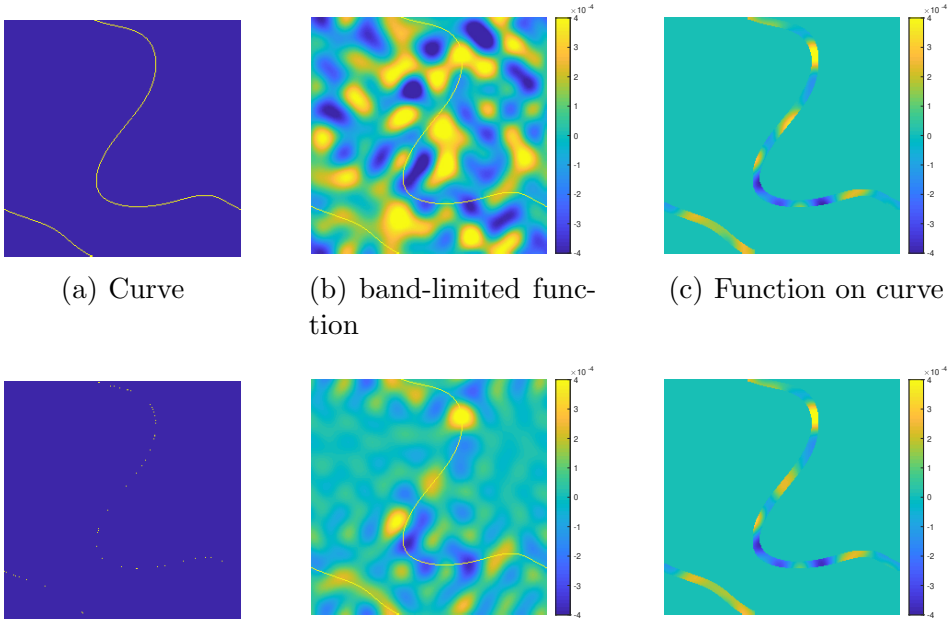
$$= \underbrace{\left[\overbrace{\boldsymbol{\beta}^T\ \Phi_\Gamma(\mathbf{a}_1)}^{\mathbf{f}(\mathbf{a}_1)}, \ldots, \overbrace{\boldsymbol{\beta}^T\ \Phi_\Gamma(\mathbf{a}_N)}^{\mathbf{f}(\mathbf{a}_N)}\right]}_{\mathbf{F}} \underbrace{\mathcal{K}(\mathbf{A})^\dagger\ \mathbf{k}_\mathbf{A}(\mathbf{x})}_{\boldsymbol{\alpha}(\mathbf{x})} \tag{3.7}$$

Here, $\mathbf{f}(\mathbf{x})$ is an $M \times 1$ vector, while $\mathbf{F}$ is an $M \times N$ matrix. $\mathcal{K}(\mathbf{A})$ is an $N \times N$ matrix and $\mathbf{k_A}(\mathbf{x})$ is an $N \times 1$ vector. Thus, if the function values at the anchor points, specified by $\mathbf{f}(\mathbf{a}_i); i = 1, \cdots, N$ are known, one can compute the function for any point $\mathbf{x} \in \mathcal{S}[\psi]$.

We note that the direct representation of a function $f : \mathbb{R}^n \to \mathbb{R}$ in (3.1) requires $|\Gamma|$ parameters, which can be viewed as the area of the green box in Fig. 2.3. By contrast, the above representation only requires $|\Gamma| \ominus |\Gamma : \Lambda|$ anchor points, which can be viewed as the area of the gray region in Fig. 2.3. The more efficient representation allows the learning of complex functions from few data points, especially in high dimensional applications.

We demonstrate the above local function representation result in a 2D setting in Fig. 3.1. Specifically, the original band-limited function is with bandwidth 13×13. The direct representation of the function has $13 \times 13 = 169$ degrees of freedom. Now, if we only care about points on a curve which is with bandwidth 3×3, then the same function living on the curve can be represented exactly using 48 anchor points, thus significantly reducing the degrees of freedom. However, note that the above representation is only exact on the curve. We note that the function goes to zero as one moves away from the curve.

The choice of anchor points depends on the geometry of the surface, including the number of irreducible components. For arbitrary training samples, we can



(a) Curve (b) band-limited function (c) Function on curve

(d) Anchor points (e) Approximation (f) Approx on curve

Figure 3.1. Illustration of the local representation of functions in 2D. We consider the local approximation of the band-limited function in (b) with a bandwidth of 13×13, living on the band-limited curve shown in (a). The bandwidth of the curve is 3×3. The curve is overlaid on the function in (b) in yellow. The restriction of the function to the vicinity of the curve is shown in (c). Our results suggest that the local function approximation requires $13^2 - 11^2 = 48$ anchor points. We randomly select the points on the curve, as shown in (d). The interpolation of the function values at these points yields the global function shown in (e). The restriction of the function to the curve in (f) shows that the approximation is good.

estimate the unknowns $\mathbf{F}$ in (3.7) from the linear relations

$$\underbrace{[\mathbf{y}_1, ..\mathbf{y}_P]}_{\mathbf{Y}} \;=\; \mathbf{F}\; \underbrace{[\boldsymbol{\alpha}(\mathbf{x}_1), \ldots, \boldsymbol{\alpha}(\mathbf{x}_P)]}_{\mathbf{Z}} \tag{3.8}$$

as $\mathbf{F} = \mathbf{Y}\mathbf{Z}^H \left(\mathbf{Z}\mathbf{Z}^H\right)^\dagger$. The above recovery is exact when we have $N = r$ achor points because $\mathbf{Z}$ has full column rank in this case. The reason why $\mathbf{Z}$ has full column rank is due to (3.4) and (3.5). Equation (3.4) suggests that $\text{rank}(\mathbf{Z}) \geq N$, while equation (3.5) shows $\text{rank}(\mathbf{Z}) \leq N$. Therefore, we have $\text{rank}(\mathbf{Z}) = N$, indicating that $\mathbf{Z}$ has

full rank in this case. When $N > r$, the $\mathbf{F}$ is obtained using the pseudo-inverse, which is based on the least square approximation.

3.2.3 Efficient computation using *kernel trick*

We use the *kernel-trick* to evaluate $\mathcal{K}(\mathbf{A})$ and $\mathbf{k}_{\mathbf{A}}(\mathbf{x})$, thus eliminating the need to explicitly evaluating the features of the anchor points and $\mathbf{x}$. Each entry of the matrix $\mathcal{K}(\mathbf{A})$ is computed as in (2.51), while the vector $\mathbf{k}_{\mathbf{A}}(\mathbf{x})$ is specified by:

$$(\mathbf{k}_{\mathbf{A}}(\mathbf{x}))_i \;=\; \underbrace{\Phi_\Gamma(\mathbf{a}_i)^H \Phi_\Gamma(\mathbf{x})}_{\kappa(\mathbf{a}_i,\mathbf{x})}, \tag{3.9}$$

which can be evaluated efficiently as nonlinear function κ (termed as kernel function) of their inner-products in $\mathbb{R}^n$. We now consider the kernel function κ for specific choices of lifting.

Using the lifting in (2.15), we obtain the kernel as

$$\kappa(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{k}\in\Gamma} \exp(j2\pi\mathbf{k}^T(\mathbf{y} - \mathbf{x})).$$

Note that the kernel is shift invariant in this setting. Since $\kappa : \mathbb{R}^n \to \mathbb{R}$ is an n dimensional function, evaluating and storing it is often challenging in multidimensional applications. We now focus on approximating the kernel efficiently for fast computation. We consider the impact of the shape of the bandwidth set Γ on the shape of the kernel. Specifically, we consider sets of the form

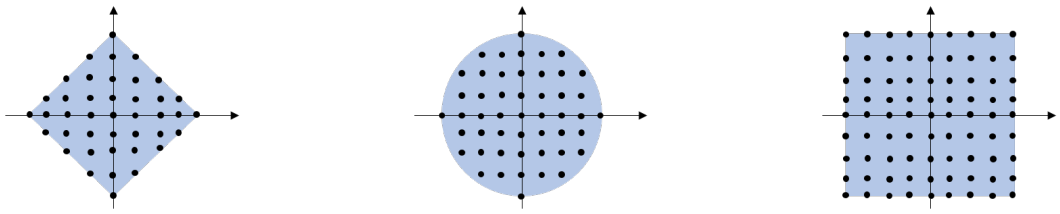
$$\Gamma = \{\mathbf{k} \in \mathbb{Z}^n, ||\mathbf{k}||_q \leq d\}, \tag{3.10}$$

where d denotes the size of the bandwidth. The integer q specifies the shape of Γ [142],

which translates to the shape of the kernel

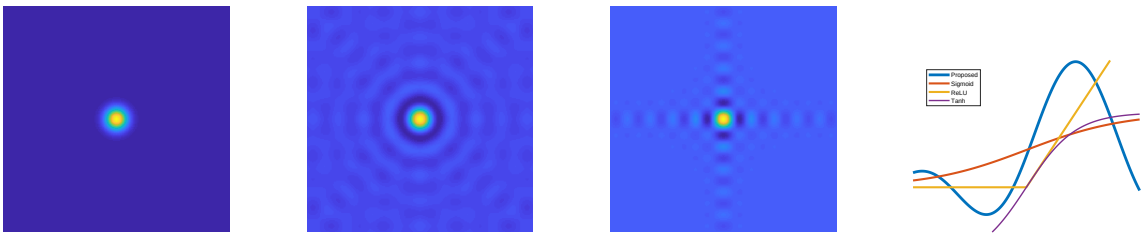
$$k^q_{d,n}(\mathbf{x}) := \sum_{\mathbf{k}\in\mathbb{Z}^n, ||\mathbf{k}||_q \leq d} \exp(j2\pi\mathbf{k}^T\mathbf{x}). \tag{3.11}$$

We term the $q = 1$ case as the diamond Dirichlet kernel. If $q = 2$, we call it the circular Dirichlet kernel. We call the Dirichlet kernel the cubic Dirichlet kernel if $q = \infty$. See Figure 3.2 for the bandwidth and Figure 3.3 to see the associated kernel.



(a) $q = 1$ (b) $q = 2$ (c) $q = \infty$

Figure 3.2. bandwidth of the set Λ with different q values.



(a) Gaussian kernel (b) Dirichlet with $q = 2$ (c) Dirichlet with $q = \infty$ (d) Plot of γ

Figure 3.3. Visualization of kernels in $\mathbb{R}^2$ and the non-linear function γ with some commonly used activation functions.

We note from the above figures that the circular Dirichlet kernel ($q = 2$) is roughly circularly symmetric, unlike the triangular or diamond kernels. This implies that we can safely approximate it as

$$\kappa(\mathbf{x}, \mathbf{y}) \approx g(\|\mathbf{x} - \mathbf{y}\|^2) \tag{3.12}$$

where $g : \mathbb{R}_+ \to \mathbb{R}$. We note that this approximation results in significantly reduced computation in the multidimensional case. The function g may be stored in a look-up table or computed analytically. We use this approach to speed up the computation of multidimensional functions in Section 3.3.

An additional simplification is to assume that $\mathbf{x}$ and $\mathbf{y}$ are unit-norm vectors. In this case, we can approximate

$$g(||\mathbf{x}_i - \mathbf{y}_i||_2^2) = g(||\mathbf{x}_i||_2^2 + ||\mathbf{y}_i||_2^2 - 2\langle \mathbf{x}_i, \mathbf{y}_i \rangle) \approx g(2 - 2\langle \mathbf{x}, \mathbf{y} \rangle) =: \gamma(\langle \mathbf{x}, \mathbf{y} \rangle), \tag{3.13}$$

where $\gamma(z) = g(1 - z/2)$. Here, we term γ as the activation function. While we do not make this simplifying assumption in our computations, it enables us to show the similarity of the computational structure of (3.6) to current neural network. The plot of this activation function, along with commonly used activation functions, is shown in Figure 3.3 (d).

With the aforementioned analysis, we can then rewrite (3.6) as

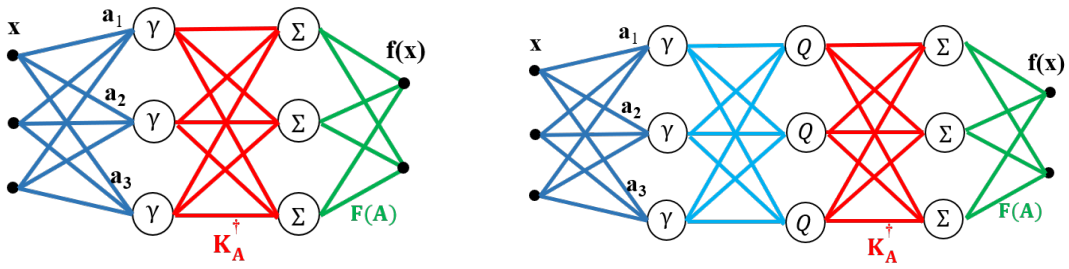
$$\mathbf{f}(\mathbf{x}) = \underbrace{[\mathbf{f}_1, \ldots, \mathbf{f}_N]}_{\mathbf{F}} \mathcal{K}(\mathbf{A})^\dagger \underbrace{\begin{bmatrix} g(\|\mathbf{x}-\mathbf{a}_1\|^2) \\ \vdots \\ g(\|\mathbf{x}-\mathbf{a}_N\|^2) \end{bmatrix}}_{\mathbf{k_A(x)}} \tag{3.14}$$

$$\approx \underbrace{\mathbf{F}\ \mathcal{K}(\mathbf{A})^\dagger}_{\widetilde{\mathbf{F}}} \underbrace{\begin{bmatrix} \gamma(\langle \mathbf{x}, \mathbf{a}_1 \rangle) \\ \vdots \\ \gamma(\langle \mathbf{x}, \mathbf{a}_N \rangle) \end{bmatrix}}_{\mathbf{\Gamma_A(x)}} \tag{3.15}$$

In the second step, we used the approximation in (3.13).

3.2.4 Optimization of the anchor points and coefficients

The above results show the existence of a computational structure of the form (3.15) with N anchor points $\mathbf{a}_1, .., \mathbf{a}_N$ on the surface and the corresponding coefficients $\tilde{\mathbf{f}_1}, .., \tilde{\mathbf{f}_N}$ that can represent the function exactly. We note that the anchor points



(a) One layer network (b) Two layers network

Figure 3.4. Computational structure of function evaluation. (a) corresponds to (3.6) to compute the band-limited multidimensional function $\mathbf{f}$ on $\mathcal{S}[\psi]$. The inner-product between the input vector $\mathbf{x}$ and the anchor templates on the surface are evaluated, followed by non-linear activation functions γ to obtain the coefficients $\alpha_i(\mathbf{x})$. These coefficients are operated with the fully connected linear layers $\mathbf{K}_\mathbf{A}^\dagger$ and $\mathbf{F}(\mathbf{A})$. The fully connected layers can be combined to obtain a single fully connected layer $\widetilde{\mathbf{F}}$. Note that this structure closely mimics a neural network with a single hidden layer. (b) uses an additional quadratic layer, which combines functions of a lower bandwidth to obtain a function of a higher bandwidth.

need not to be selected as a subset of the training data. We note that Corollary 22 guarantees $\mathcal{K}(\mathbf{A})$ to have full column rank as $N = r$. However, the condition number of this matrix may be poor, depending on the choice of the anchor points. It may be worthwhile to choose the anchors such that the condition number of $\mathcal{K}(\mathbf{A})$ is low, which will reduce the noise amplification in (3.5).

We hence propose to solve for the anchor points $\mathbf{A}$ and the corresponding coefficients $\widetilde{\mathbf{F}}$ such that it minimizes the least square error evaluated on the training data:

$$\widetilde{\mathbf{F}}^*, \mathbf{A}^* = \arg\min_{\widetilde{\mathbf{F}}, \mathbf{A}} \sum_{i=1}^{N_{\text{train}}} \|\widetilde{\mathbf{F}}\ \mathbf{\Gamma}_{\mathbf{A}}(\mathbf{x}_i) - \mathbf{y}_i\|^2 \tag{3.16}$$

We propose to minimize the above expression using stochastic gradient descent to simultaneously derive the anchor points $\mathbf{a}_1, .., \mathbf{a}_N$ as well as the coefficients, which are the learnable parameters of the single layer network. Specifically, we consider noisy patches as inputs and noise-free pixels as the desired outputs; the parameters of the network are then obtained by minimizing (3.16).

3.3 Relation to neural networks

We now briefly discuss the close relation of the proposed framework with neural networks. We consider the function learning setting, which is considered in Section 3.2 and show that the computational structure closely mimics a neural network with one hidden layer. We discuss briefly the benefits of depth in improving the representation. We also show that the above framework can be used to approximate the learning of a manifold from data, which can be viewed as a signal subspace alternative to the null-space approach considered in Section 2.5. We also show that the computational

structure closely mimics an auto-encoder.

3.3.1 Task/function learning from input output pairs

We now focus on the learning of a function (3.14) from training data pairs and will show its equivalence with neural networks. Note that the computation involves the inner product of the input signal $\mathbf{x}$ with templates $\mathbf{a}_i; i = 1, .., N$, followed by the non-linear activation function γ to obtain $\mathbf{k}_{\mathbf{A}}(\mathbf{x})$. These terms are then weighted by the fully connected layer $\mathcal{K}\left(\mathbf{A}\right)^{\dagger}$, followed by weighting by the second fully connected layer $\widetilde{\mathbf{F}}$. See Fig. 3.4 for the visual illustration.

As noted above, the representation using anchor points to reduce the degrees of freedom significantly compared to the direct representation. However, we note that the number of parameters needed to represent a high bandwidth function in high dimensions is still high. We now provide some intuition on how the low-rank tensor approximation of functions and composition can explain the benefit of common operations in deep networks.

We now consider the case when the band-limited multidimensional function $f : \mathbb{R}^n \to \mathbb{R}$ in (3.1) can be approximated as

$$f(\mathbf{x}) = \Big(\sum w_i \; f_i(\mathbf{x})\Big)^2. \tag{3.17}$$

Clearly, the bandwidth of f is almost twice that of $f_i : \mathbb{R}^n \to \mathbb{R}$, showing the benefit of adding layers. While an arbitrary function with the same bandwidth as f cannot be represented as in (3.17), one may be able to approximate it closely. The new layer will have a quadratic non-linearity Q, if the function has the form (3.17). Note that one may use arbitrary non-linearity in place of the quadratic one in (3.17).

Similarly, one may perform a low-rank tensor approximation of an arbitrary N dimensional function $f : \mathbb{R}^n \to \mathbb{R}$. Specifically, the approximation involves the sum of products of 1-D functions.

$$f(x_1, .., x_N) \approx \sum_{i=1}^{r} h_1^{(i)}(x_1) \cdot \ h_2^{(i)}(x_2) \ldots \ h_N^{(i)}(x_N), \tag{3.18}$$

where $h_i : \mathbb{R} \to \mathbb{R}$. The above sum of products can also be realized by taking weighted linear combination of 1-D functions, followed by a non-linearity as in (3.17). This allows one to have a hierarchical structure, where lower dimensional functions are pooled together to represent a multidimensional function.

In image processing applications, the functions to be learned are shift-invariant. This allows one to learn functions of small image patches (e.g. 3×3) of a specified dimension at each layer. The functions on nearby pixels in the output thus correspond to information from different 3×3 neighborhoods. The low-dimensional functions from non-overlapping 3×3 neighborhoods could be combined with downsampling as in (3.18) to represent a high dimensional function (e.g. 9×9) neighborhoods. The process can be repeated to improve the efficiency of representation.

3.3.2 Relation to auto-encoders

We note that the space of band-limited functions of the form (3.1) can reasonably approximate lower order polynomials in $\mathbb{R}^n$ for sufficiently high bandwidth Γ [123]. In particular, let us assume that there exists a set of coefficients $\boldsymbol{\beta}$ such that

$$\mathbf{x} \approx \tilde{\mathbf{x}} = \sum_{\mathbf{k} \in \Gamma} \beta_{\mathbf{k}} \exp(j2\pi \mathbf{k}^T \mathbf{x}) \tag{3.19}$$

In this case, the above results imply that one can represent any point on the surface $\mathcal{S}[\psi]$ as

$$\mathbf{x} \approx \underbrace{[\mathbf{a}_1, .., \mathbf{a}_n]}_{\mathbf{A}} \underbrace{\mathcal{K}(\mathbf{A})^\dagger \ \mathbf{k}_{\mathbf{A}}(\mathbf{x})}_{\boldsymbol{\alpha}(\mathbf{x})} \tag{3.20}$$

We note that the resulting network is hence essentially an auto-encoder. Specifically, the inner-products between the feature vectors of $\mathbf{x}$ and the anchor point $\mathbf{a}_i$ denoted by $\alpha(\mathbf{x})$ can be viewed as the latent features or compact code. As described previously, the coefficients $\boldsymbol{\alpha} = \mathcal{K}(\mathbf{A})^\dagger \mathbf{k}_{\mathbf{A}}(\mathbf{x})$ captures the geometry of the surface, while the top layer $\mathbf{A}$ is the decoder that recover the signal from its latent vectors.

We note that the surface recovery algorithms in Section 2.5 follow a null-space approach, where we identify the null-space of the feature space or equivalently the annihilation functions from the samples of the surface. Specifically, the sum of squares of the null-space functions in Section 2.5.2 provides a measure of the error in projecting the feature vector to the null-space of the feature matrix.

$$\gamma(\mathbf{x}) = \sum_{i=1}^{|\Gamma \ominus \Lambda|} |\mu_i(\mathbf{x})|^2 = \sum_{i=1}^{|\Gamma \ominus \Lambda|} |\mathbf{n}_i^T \Phi_\Gamma(\mathbf{x})|^2 \tag{3.21}$$

$$= \|\mathbf{N} \ \Phi_\Gamma(\mathbf{x})\|^2 \tag{3.22}$$

where $\mathbf{n}_i$ are the null-space vectors. The projection energy is zero if the point $\mathbf{x}$ is on $\mathcal{S}$ and is high when it is far from it.

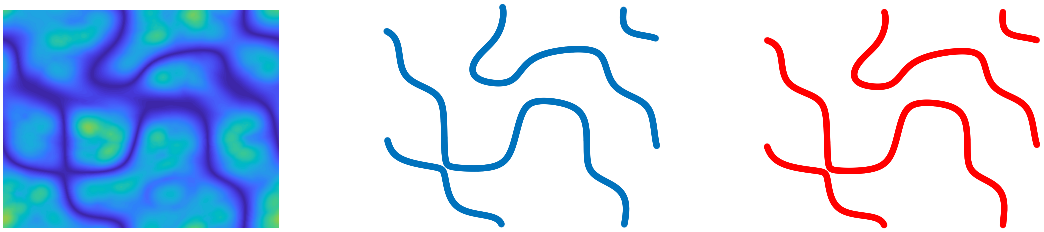
By contrast, the auto-encoder approach can be viewed as a signal subspace approach, where we project the samples to the basis vectors specified by the feature vectors of the anchors $\Phi_\Gamma(\mathbf{a}_i)$. Specifically, we use the non-linearity specified by (3.13) and trained the network parameters ($\mathbf{A}$ as well as the weights of the inner-products)

using stochastic gradient descent. The training data corresponds to randomly drawn points on the surface. To ensure that the network learns a projection, we trained the network as a denoising auto-encoder; the inputs correspond to samples on the surface corrupted with Gaussian noise, while the labels are the true samples. Once the training is complete, we plot the approximation error

$$E(\mathbf{x}) = \|\mathbf{x} - \mathbf{F}\mathcal{K}(\mathbf{A})^{\dagger}\mathbf{k}_{\mathbf{A}}(\mathbf{x})\|^2 = \| \underbrace{\left(\mathbf{I} - \mathbf{F}\mathcal{K}(\mathbf{A})^{\dagger}\mathbf{k}_{\mathbf{A}}\right)}_{\mathcal{R}}(\mathbf{x})\|^2 \tag{3.23}$$

as a function of the input point in Fig. 3.5.

We trained the network using the exemplar curve shown in Fig. 2.16. We randomly choose 1000 points on the curve as the training data and 250 features are chosen in the middle layer. The bandwidth of the Dirichlet kernel is chosen to be 15. The trained network is then used to learn the curve. The learned results are shown in Fig. 3.5. From which one can see that the proposed learning framework performs well. We note that the projection error is close to zero on the surface, while it is high if it is away from the surface. Note that this closely mimics the plot in Fig. 2.16. Once trained, the surface can be estimated in low-dimensional settings as the zero set of the projection error as shown in Fig. 3.5.(b), which closely approximates the true curve in (c). We note that $\mathcal{R}$ can be viewed as a residual denoising auto-encoder. Once trained, this network can be used as a prior in inverse problems as in [3], where we have used the null-space network in Section 3.3. We have also used the null-space prior (3.21) in our prior work [100], where the null-space basis was learned as described in Section 2.6.



(a) Learned curve (b) Contour line (c) Original curve

Figure 3.5. Illustration of the surface learning network using the curve in Fig. 2.16. (a) and (b) are the learned results. We compared the learned curve (blue curve) with the original curve (red curve) in (c). From which we see that the two curves are almost the same, indicating that the learned network performs well.

3.4 Illustration in denoising

We now illustrate the preliminary utility of the proposed network in image denoising. Specifically, we consider the learning of a function $f : \mathbb{R}^{p^2} \to \mathbb{R}$, which predicts the denoised center pixel of a patch from the noisy $p \times p$ patch. The function f in p^2 dimensional space is associated with a large number of free parameters; learning of these unknowns are challenging due to the curse of dimensionality. Then the result in the previous section offers a work-around, which suggests that the function can be expressed as the linear combination of the features of "anchor-patches", weighted by $\mathbf{p}$.

We propose to learn the anchor patches $\mathbf{a}_i$ and the function values $f(\mathbf{a}_i)$ from exemplar data using stochastic gradient descent to minimize (3.16). Note that the learned representation is valid for any patch, and hence the proposed scheme is essentially a convolutional neural network. The difference of our structure in (3.15) with the commonly used convolutional neural networks (CNN) structure is the activation

function γ. We replaced the ReLU non-linearity in a network with the proposed function γ in a single layer network. For the two-layer network, we replaced the ReLU non-linearity with γ and Q as indicated in (3.17).

We first tested the performance of the network on the MNIST dataset [62]. In the experiments, we choose the patch size to be 7×7 and $d = 7$ in (3.11). We also trained a ReLU network with the same parameters for comparison. Besides, we compared the proposed scheme against non-local means (NLM) and dictionary learning (DL) [35]. All algorithms, except for NLM were trained using the MNIST training set provided in TensorFlow. For the proposed network and the ReLU network, they are trained using 300 epoches and for the dictionary learning method, 500 iterations are used to learn the dictionaries. The comparison of the testing results is shown in Figure 3.6. The comparison of the PSNR is reported in the caption. The results show that the neural network based approaches offer improved performance compared to dictionary learning and non-local methods. Our results also show that the proposed networks provide comparable, if not slightly better performance, compared to the ReLU networks. The results also show the slight improvement in performance offered by the proposed two-layer networks over single layer networks.

The size of the image in the MNIST dataset is small. To better demonstrate the performance of the proposed network, we also applied the proposed scheme to the denoising of natural images. The algorithm was trained on the images of Hill, Cameraman, Couple, Bridge, Barbara and Boat at three different noise settings. We assume the noise is Gaussian white noise in the natural images setting. We compared

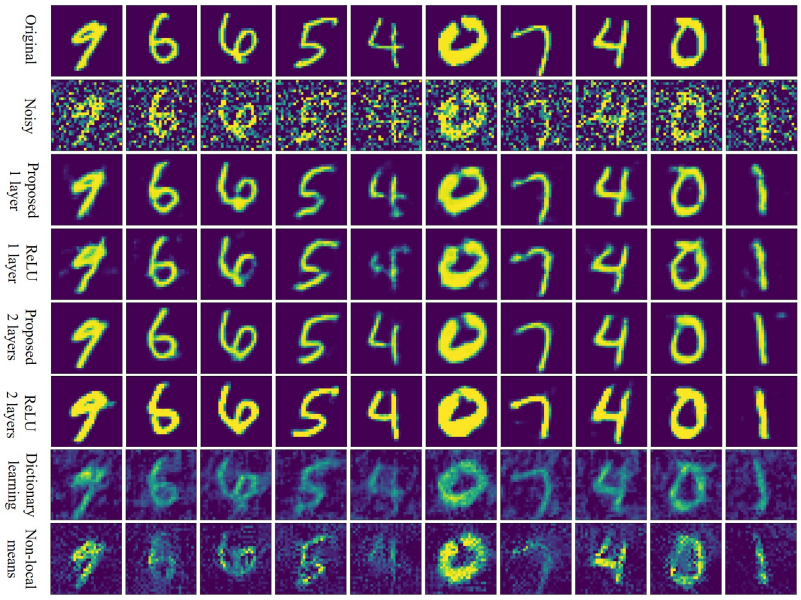


Figure 3.6. Comparison of our learned denoiser using the proposed activation function and the ReLU activation function. The testing results show that the denoising performance using the proposed activation function is comparable to the performance using ReLU. The eight rows in the figure correspond to the original images, the noisy images, the denoised images using the proposed one-layer network, the denoised images using one layer ReLU network, the denoised images using the proposed two-layer network, the denoised images using two-layer ReLU network, the denoised images using dictionary learning and the denoised images using non-local means. The averaged PSNR of the denoised images using the proposed one-layer network, one layer ReLU network, proposed two-layer network, two-layer ReLU network, dictionary learning and non-local means are 19.68 dB, 20.03 dB, 20.86 dB, 17.48 dB, 14.76 dB and 14.28 dB respectively. From the quantitative results, we can see that our proposed one-layer network performs comparable to the one-layer ReLU network. For the proposed two-layer network, the performance is getting better from both quantitative and visual points of view. For the two-layer ReLU network, visually the performance is better than that of the one-layer ReLU network. But the PSNR is getting worse. The main reason that causes the low PSNR for the two-layer ReLU network is the change of the pixel values on each hand-written digit.

the proposed scheme against dictionary learning (DL), non-local means (NLM) and transform learning (TL) [107]. In the experiments for natural images, the patch size is chosen as 9×9 and $d = 7$ in (3.11). For the proposed network and the ReLU

network, they are trained using 300, 400, 450 epoches corresponding to the noise level $\sigma = 10, 20, 100$, and for the dictionary learning method, 500 iterations are used to learn the dictionaries. We then tested the denoising performance on two natural images: Man and Lighthouse.

The quantitative results (PSNR) of the algorithm are shown in Table 3.1, while the results on Man and Lighthouse with noise of standard deviation $\sigma = 20$ are shown in Fig. 3.7 and Fig. 3.8. In Table 3.1, Fig. 3.7 and Fig. 3.8, "ReLU1" and "ReLU2" represent one-layer ReLU network and two-layer ReLU network, while "Proposed1" and "Proposed2" stand for the proposed one-layer network and proposed two-layer network. The results show that the performance of the neural network schemes is superior to classical methods and the proposed networks provide comparable or slightly better performance than the ReLU networks.

Img.	σ	DL	NLM	TL	ReLU1	ReLU2	Proposed1	Proposed2
Man	10	26.63	26.64	27.41	30.29	31.11	30.99	**31.19**
	20	26.11	26.35	27.02	27.47	27.33	27.25	**27.63**
	100	19.69	20.95	21.65	21.85	**22.11**	21.91	22.06
Lighthouse	10	27.08	29.08	28.71	28.88	29.27	30.05	**30.28**
	20	25.51	25.21	25.92	26.25	26.33	26.69	**26.74**
	100	19.14	20.14	20.15	20.21	20.46	20.35	**20.47**

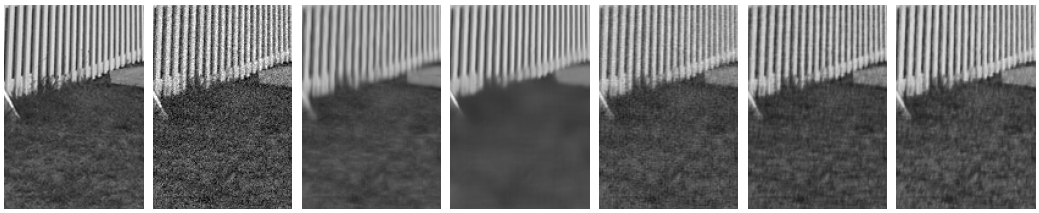
Table 3.1. The PSNR (dB) of the denoised results for the two testing natural images with different noise level.

(a) Orig. (b) Noisy (c) DL (d) NLM (e) TL (f) ReLU2 (g) Prop.

(h) Orig. (i) Noisy (j) DL (k) NLM (l) TL (m) ReLU2 (n) Prop.

Figure 3.7. Comparison of the proposed denoising algorithms on the image "Man" with $\sigma = 20$.

(a) Orig. (b) Noisy (c) DL (d) NLM (e) TL (f) ReLU2 (g) Prop.



(h) Orig. (i) Noisy (j) DL (k) NLM (l) TL (m) ReLU 2 (n) Prop.

Figure 3.8. Comparison of the proposed denoising algorithms on the image "Lighthouse" with $\sigma = 20$.

3.5 Conclusion

In this work, we considered a data model, where the signals are localized to a surface that is the zero level set of a band-limited function ψ. The bandwidth

of the function can be seen as a complexity measure of the surface. We show that the non-linear features of the samples, obtained by an exponential lifting, satisfy an annihilation relation. Using the annihilation relation, we developed theoretical sampling guarantees for the unique recovery of the surface. Our main contribution here is to prove that with probability 1, the surface can be uniquely recovered using a collection of samples, whose number is equal to the degrees of freedom of the representation. When the true bandwidth of the surface is unknown, which is usually the case, we introduced a method using the SoS polynomial to specify the surface. We also introduced the way to get back the samples when the original samples are corrupted by noise.

We then use this model to efficiently represent arbitrary band-limited functions f living on the surface. We show that the exponential features of the points on the surface live in a low-dimensional subspace. This subspace structure is used to represent the f efficiently using very few parameters. We note that the computational structure of the function evaluation mimics a single-layer neural network. We applied the proposed computational structure to the context of image denoising.

CHAPTER 4

MODEL BASED COMPUTATIONAL IMAGING USING UNION OF SURFACES PRIOR: DEEP GENERATIVE STORM MODEL

4.1 Introduction

The imaging of time-varying objects at high spatial and temporal resolution is key to several modalities, including MRI and microscopy. A central challenge is the need for high resolution in both space and time [66, 127]. Several computational imaging strategies have been introduced in MRI to improve the resolution, especially in the context of free-breathing and ungated cardiac MRI. A popular approach pursued by several groups is self-gating, where cardiac and respiratory information is obtained from central k-space regions (navigators) using bandpass filtering or clustering [19,24,40,41,103]. The data is then binned to the respective phases and recovered using total variation or other priors. Recently, approaches using smooth manifold regularization have been introduced. These approaches model the images in the time series as points on a high-dimensional manifold [5, 80, 81, 97, 100]. Manifold regularization algorithms, including the smoothness regularization on manifolds (SToRM) framework [5, 97, 100], have shown good performance in several dynamic imaging applications. Since the data is not explicitly binned into specific phases as in the self-gating methods, manifold algorithms are less vulnerable to clustering errors than navigator-based corrections. Despite the benefits, a key challenge with the current manifold methods is the high memory demand. Unlike self-gating methods that only recover specific phases, manifold methods recover the entire time series. The limited

memory on current GPUs restricts the number of frames that can be recovered simultaneously, which makes it challenging to extend the model to higher dimensionalities. The high memory demand also makes it difficult to use spatial regularization priors on the images using deep learned models.

Our main focus is to capitalize on the power of deep convolutional neural networks (CNN) to introduce a memory efficient generative or synthesis formulation of SToRM. CNN based approaches are now revolutionizing image reconstruction, offering significantly improved image quality and fast image recovery [30, 55, 78, 102, 136, 137, 144]. In the context of MRI, several novel approaches have been introduced [138, 139], including transfer-learning [29], domain adaptation [48], learning-based dynamic MRI [111], and generative-adversarial models [27, 28, 146]. Unlike many CNN-based approaches, the proposed scheme does not require pre-training using large amounts of training data. This makes the approach desirable in free-breathing applications, where the acquisition of fully sampled training data is infeasible. We note that the classical SToRM approach can be viewed as an analysis regularization scheme (see Fig. 4.1.(a)). Specifically, a non-linear injective mapping is applied on the images such that the mapped points of the alias-free images lie on a low-dimensional subspace [100, 153, 155]. When recovering images from undersampled data, the nuclear norm prior is applied in the transform domain to encourage their non-linear mappings to lie in a subspace. Unfortunately, this analysis approach requires the storage of all the image frames in the time series, which translates to high memory demand. The proposed generative SToRM formulation offers quite sig-

nificant compression of the data, which can overcome the above challenge. Both the relation between the analysis and synthesis formulations and the relation of the synthesis formulation to neural networks were established in earlier work [153].

We assume that the image volumes in the dataset are smooth non-linear functions of a few latent variables, i.e., $\mathbf{x}_t = \mathcal{G}_\theta(\mathbf{z}_t)$, where $\mathbf{z}_t$ are the latent vectors in a low-dimensional space. $\mathbf{x}_t$ is the t-th generated image frame in the time series. This explicit formulation implies that the image volumes lie on a smooth non-linear manifold in a high-dimensional ambient space (see Fig. 4.1.(b)). The latent variables capture the differences between the images (e.g., cardiac phase, respiratory phase, contrast dynamics, subject motion). We model the $\mathcal{G}$ using a CNN, which offers a significantly compressed representation. Specifically, the number of parameters required by the model (CNN weights and latent vectors) are several orders of magnitude smaller than required for the direct representation of the images. The compact model proportionately reduces the number of measurements needed to recover the images. In addition, the compression also enables algorithms with much smaller memory footprint and computational complexity. We propose to jointly optimize for the network parameters θ and the latent vector $\mathbf{z}_t$ based on the given measurements. The smoothness of the manifold generated by $\mathcal{G}_\theta(\mathbf{z})$ depends on the gradient of $\mathcal{G}_\theta$ with respect to its input. To enforce the learning of a smooth image manifold, we regularize the norm of the Jacobian of the mapping $\|J_z\mathcal{G}_\theta\|^2$. We experimentally observe that by penalizing the gradient of the mapping, the network is encouraged to learn meaningful mappings. Similarly, the images in the time series are expected to vary smoothly

in time. Hence, we also use a Tikhonov smoothness penalty on the latent vectors $\mathbf{z}_t$ to further constrain the solutions. We use the ADAM optimizer with stochastic gradients, where random batches of $\mathbf{z}_i$ and $\mathbf{b}_i$ are chosen at iteration to determine the parameters. Unlike traditional CNN methods that are fast during testing/inference, the direct application of this scheme to the dynamic MRI setting is computationally expensive. We use approximations, including progressive-in-time optimization and an approximated data term that avoids non-uniform fast Fourier transforms, to significantly reduce the computational complexity of the algorithm.

The proposed approach is inspired by deep image prior (DIP), which was introduced for static imaging problems [131], as well as its extension to dynamic imaging [54]. The key difference of the proposed formulation is the joint optimization of the latent variables $\mathbf{z}$ and $\mathcal{G}$. The work of Jin ea tl. [54] was originally developed for CINE MRI, where the latent variables were obtained by linearly interpolating noise variables at the first and last frames. Their extension to real-time applications involved setting noise latent vectors at multiples of a preselected period, followed by linearly interpolating the noise variables. This approach is not ideally suited for applications with free breathing, when the motion is not periodic. Another key distinction is the use of regularization priors on the network parameters and latent vectors, which encourages the mapping to be an isometry between latent and image spaces. Unlike DIP methods, the performance of the network does not significantly degrade with iterations. While we call our algorithm "generative SToRM", we note that our goal is not to generate random images from stochastic inputs as in generative-

adversarial networks (GAN). In particular, we do not use adversarial loss functions where a discriminator is jointly learned as in the literature [15].

4.2 Background

4.2.1 Dynamic MRI from undersampled data: problem setup

Our main focus is to recover a series of images $\mathbf{x}_1, ..\mathbf{x}_M$ from their undersampled multichannel MRI measurements. The multidimensional dataset is often compactly represented by its Casoratti matrix

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & ... & \mathbf{x}_M \end{bmatrix}. \tag{4.1}$$

Each of the images is acquired by different multichannel measurement operators

$$\mathbf{b}_i = \mathcal{A}_i(\mathbf{x}_i) + \mathbf{n}_i, \tag{4.2}$$

where $\mathbf{n}_i$ is zero mean Gaussian noise matrix that corrupts the measurements.

4.2.2 Smooth manifold models for dynamic MRI

The smooth manifold methods model images $\mathbf{x}_i$ in the dynamic time series as points on a smooth manifold $\mathcal{M}$. These methods are motivated by continuous domain formulations that recover a function f on a manifold from its measurements as

$$f = \arg\min_f \sum_i \|f(\mathbf{x}_i) - \mathbf{b}_i\|^2 + \lambda \int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f\|^2 d\mathbf{x} \tag{4.3}$$

where the regularization term involves the smoothness of the function on the manifold.

This problem is adapted to the discrete setting to solve for images lying on a smooth manifold from its measurements as

$$\mathbf{X} = \arg\min_{\mathbf{X}} \sum_{i=1}^{M} \|\mathcal{A}(\mathbf{x}_i) - \mathbf{b}_i\|^2 + \lambda \ \text{trace}(\mathbf{X}\mathbf{L}\mathbf{X}^H), \tag{4.4}$$

where $\mathbf{L}$ is the graph Laplacian matrix. $\mathbf{L}$ is the discrete approximation of the Laplace-Beltrami operator on the manifold, which depends on the structure or geometry of the manifold. The manifold matrix $\mathbf{L}$ is estimated from k-space navigators. Different approaches, ranging from proximity-based methods [97] to kernel low-rank regularization [100] and sparse optimization [80], have been introduced.

The results of earlier work [100, 155] show that the above manifold regularization penalties can be viewed as an analysis prior. In particular, these schemes rely on a fixed non-linear mapping φ of the images. The theory shows that if the images $\mathbf{x}_1, ..\mathbf{x}_M$ lie in a smooth manifold/surface or union of manifolds/surfaces, the mapped points live on a subspace or union of subspaces. The low-dimensional property of the mapped points $\varphi(\mathbf{x}_1), ..\varphi(\mathbf{x}_M)$ is used to recover the images from undersampled data or derive the manifold using a kernel low-rank minimization scheme:

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} \sum_{i=1}^{M} \|\mathcal{A}(\mathbf{x}_i) - \mathbf{b}_i\|^2 + \lambda \parallel [\varphi(\mathbf{x}_1), .., \varphi(\mathbf{x}_N)] \parallel_*. \tag{4.5}$$

This nuclear norm regularization scheme is minimized using an iterative reweighted algorithm, whose intermediate steps match (4.4). The non-linear mapping φ may be viewed as an analysis operator that transforms the original images to a low-dimensional latent subspace, very similar to analysis sparsity-based approaches used in compressed sensing.

4.2.3 Unsupervised learning using Deep Image Prior

The recent work of DIP uses the structure of the network as a prior [131], enabling the recovery of images from ill-posed measurements without any training data. Specifically, DIP relies on the property that CNN architectures favor image

data more than noise. The regularized reconstruction of an image from undersampled and noisy measurements is posed in DIP as

$$\{\boldsymbol{\theta}^*\} = \arg\min_{\boldsymbol{\theta}} \|\mathcal{A}(\mathbf{x}) - \mathbf{b}\|^2 \quad \text{such that} \quad \mathbf{x} = \mathcal{G}_{\boldsymbol{\theta}}[\mathbf{z}] \tag{4.6}$$

where $\mathbf{x} = \mathcal{G}_{\boldsymbol{\theta}^*}(\mathbf{z})$ is the recovered image, generated by the CNN generator $\mathcal{G}_{\boldsymbol{\theta}^*}$ whose parameters are denoted by $\boldsymbol{\theta}$. Here, $\mathbf{z}$ is the random latent variable, which is chosen as random noise and kept fixed.

The above optimization problem is often solved using stochastic gradient descent (SGD). Since CNNs are efficient in learning natural images, the solution often converges quickly to a good image. However, when iterated further, the algorithm also learns to represent the noise in the measurements if the generator has sufficient capacity, resulting in poor image quality. The general practice is to rely on early termination to obtain good results. This approach was recently extended to the dynamic setting by Jin et al. [54], where a sequence of random vectors was used as the input.

4.3 Deep generative SToRM model

We now introduce a synthesis SToRM formulation for the recovery of images in a time series from undersampled data (see Fig. 2.2.(b)). Rather than relying on a non-linear mapping of images to a low-dimensional subspace [100] (see Fig. 4.1.(a)), we model the images in the time series as non-linear functions of latent vectors living in a low-dimensional subspace.

4.3.1 Generative model

We model the images in the time series as

$$\mathbf{x}_i = \mathcal{G}_\theta(\mathbf{z}_i), i = 1, .., M, \tag{4.7}$$

where $\mathcal{G}_\theta$ is a non-linear mapping, which is termed as the generator. Inspired by the extensive work on generative image models [6, 45, 131], we represent $\mathcal{G}_\theta$ by a deep CNN, whose weights are denoted by θ. The parameters $\mathbf{z}_i$ are the latent vectors, which live in a low-dimensional subspace. The non-linear mapping $\mathcal{G}_\theta$ may be viewed as the inverse of the image-to-latent space mapping φ, considered in the SToRM approach.

We propose to estimate the parameters of the network θ as well as the latent vectors $\mathbf{z}_i$ by fitting the model to the undersampled measurements. The main distinction of our framework with DIP, which is designed for a single image, is that we use the same generator for all the images in the dynamic dataset. The latent vector $\mathbf{z}_i$ for each image is different and is also estimated from the measurements. This strategy allows us to exploit non-local information in the dataset. For example, in free-breathing cardiac MRI, the latent vectors of images with the same cardiac and respiratory phase are expected to be similar. When the gradient of the network is bounded, the output images at these time points are expected to be the same. The proposed framework is hence expected to learn a common representation from these time-points, which are often sampled using different sampling trajectories. Unlike conventional manifold methods [80, 97, 100], the use of the CNN generator also offers spatial regularization.

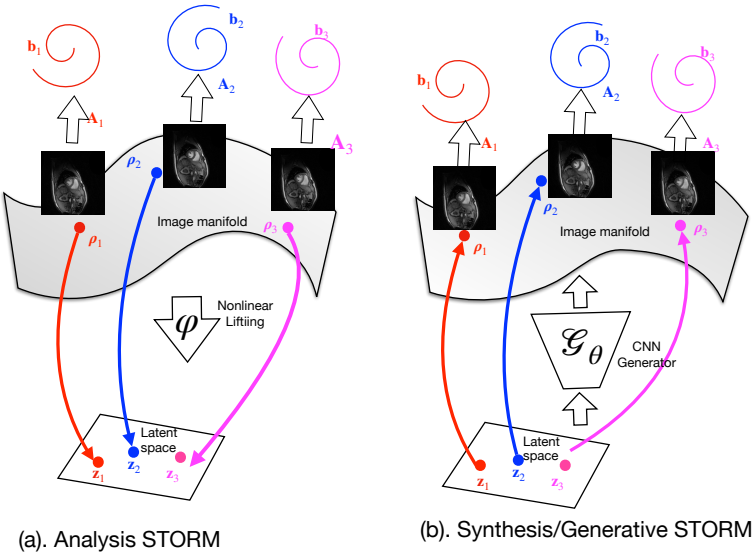


Figure 4.1. Illustration of (a) analysis SToRM and (b) generative SToRM. Analysis SToRM considers a non-linear (e.g. exponential) lifting of the data. If the original points lie on a smooth manifold, the lifted points lie on a low-dimensional subspace. The analysis SToRM cost function in (4.5) is the sum of the fit of the recovered images to the undersampled measurements and the nuclear norm of the lifted points. A challenge with analysis SToRM is its high memory demand and the difficulty in adding spatial regularization. The proposed method models the images as the non-linear mapping $\mathcal{G}_\theta$ of some latent vectors $\mathbf{z}_i$, which lie in a very low-dimensional space. Note that the same generator is used to model all the images in the dataset. The number of parameters of the generator and the latent variables is around the size of a single image, which implies a highly compressed representation. In addition, the structure of the CNN offers spatial regularization as shown in DIP. The proposed algorithm in (4.13) estimates the parameters of the generator and the latent variables from the measured data. A distance regularization prior is added to the generator to ensure that nearby points in the latent subspace are mapped to nearby points on the manifold. Similarly, a temporal regularization prior is added to the latent variables. The optimization is performed using ADAM with batches of few images.

It is often impossible to acquire fully-sampled training data in many free-breathing dynamic imaging applications, and a key benefit of this framework over conventional neural network schemes is that no training data is required. As discussed previously, the number of parameters of the model in (4.7) is orders of magnitude

smaller than the number of pixels in the dataset. The dramatic compression offered by the representation, together with the mini-batch training provides a highly memory-efficient alternative to current manifold based and low-rank/tensor approaches. Although our focus is on establishing the utility of the scheme in 2-D settings in this chapter, the approach can be readily translated to higher dimensional applications. Another benefit is the implicit spatial regularization brought in by the convolutional network as discussed for DIP. We now introduce novel regularization priors on the network and the latent vectors to further constrain the recovery to reduce the need for manual early stopping.

4.3.2 Distance/Network regularization

As in the case of analysis SToRM regularization [97, 100], our interest is in generating a manifold model that preserves distances. Specifically, we would like the nearby points in the latent space to map to similar images on the manifold. With this interest, we now study the relation between the Euclidean distances between their latent vectors and the shortest distance between the points on the manifold (geodesic distance).

We consider two points $\mathbf{z}_1$ and $\mathbf{z}_2$ in the latent space, which are fed to the generator to obtain $\mathcal{G}(\mathbf{z}_1)$ and $\mathcal{G}(\mathbf{z}_2)$, respectively. We have the following result, which relates the the Euclidean distance $\|\mathbf{z}_1-\mathbf{z}_2\|^2$ to the geodesic distance $\text{dist}_{\mathcal{M}}\left(\mathcal{G}(\mathbf{z}_1), \mathcal{G}(\mathbf{z}_2)\right)$, which is the shortest distance on the manifold. The setting is illustrated in Fig. 4.2, where the geodesic distance is indicated by the red curve.

Proposition 23. *Let $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^n$ be two nearby points in the latent space, with*

mappings denoted by $\mathcal{G}(\mathbf{z}_1), \mathcal{G}(\mathbf{z}_2) \in \mathcal{M}$. *Here,* $\mathcal{M} = \{G(\mathbf{z}) | \mathbf{z} \in \mathbb{R}^n\}$. *Then, the geodesic distance on the manifold satisfies:*

$$\text{dist}_{\mathcal{M}}\big(\mathcal{G}(\mathbf{z}_1), \mathcal{G}(\mathbf{z}_2)\big) \leq \|\mathbf{z}_1 - \mathbf{z}_2\|_F \; \|J_z\big(\mathcal{G}\left(\mathbf{z}_1\right)\big)\|_F. \tag{4.8}$$

Proof. The straight-line between the latent vectors is denoted by $c(s), s \in [0, 1]$ with $c(0) = \mathbf{z}_1$ and $c(1) = \mathbf{z}_2$. We also assume that the line is described in its curvilinear abscissa, which implies $\|c'(s)\| = 1; \forall s \in [0, 1]$. We note that $\mathcal{G}$ may map to the black curve, which may be longer than the geodesic distance. We now compute the length of the black curve $\mathcal{G}[c(s)]$ as

$$d = \int_0^1 \|\nabla_s \mathcal{G}\left[c(s)\right]\| ds. \tag{4.9}$$

Using the chain rule and denoting the Jacobian matrix of $\mathcal{G}$ by J_z, we can simplify the above distance as

$$\begin{aligned} d &= \int_0^1 \|J_z\left(\mathcal{G}\right) c'(s)\|_F ds \\ &\leq \int_0^1 \|J_z\left(\mathcal{G}\right)\|_F \underbrace{\|c'(s)\|_F}_{1} ds \\ &= \|J_z\left(\mathcal{G}[\mathbf{z}_1]\right)\|_F \underbrace{\int_0^1 ds}_{\|\mathbf{z}_1 - \mathbf{z}_2\|}. \end{aligned} \tag{4.10}$$

We used the Cauchy-Schwartz inequality in the second step and in the last step, we use the fact that $J_z\mathcal{G}\left(c(t)\right) = J_z\mathcal{G}\left(\mathbf{z}_1\right) + \mathcal{O}(t)$ when the points $\mathbf{z}_1$ and $\mathbf{z}_2$ are close. Since the geodesic distance is the shortest distance on the manifold, we have $\text{dist}_{\mathcal{M}}\big(\mathcal{G}(\mathbf{z}_1), \mathcal{G}(\mathbf{z}_2)\big) \leq d$ and hence we obtain (4.8). □

The result in (4.8) shows that the Frobenius norm of the Jacobian matrix $\|J_z\mathcal{G}\|$ controls how far apart $\mathcal{G}$ maps two vectors that are close in the latent space.

We would like points that are close in the latent space map to nearby points on the manifold. We hence use the gradient of the map:

$$R_{\text{distance}} = \|J_z\big(\mathcal{G}(\mathbf{z})\big)\|_F^2 \tag{4.11}$$

as a regularization penalty. We note that the above penalty will also encourage the learning of a mapping $\mathcal{G}$ such that the length of curve $\mathcal{G}(c(t))$ is the geodesic distance. We note that the above penalty can also be thought of as a network regularization. Similar gradient penalties are used in machine learning to improve generalization ability and to improve the robustness to adversarial attacks [133]. The use of gradient penalty is observed to be qualitatively equivalent to penalizing the norm of the weights of the network.

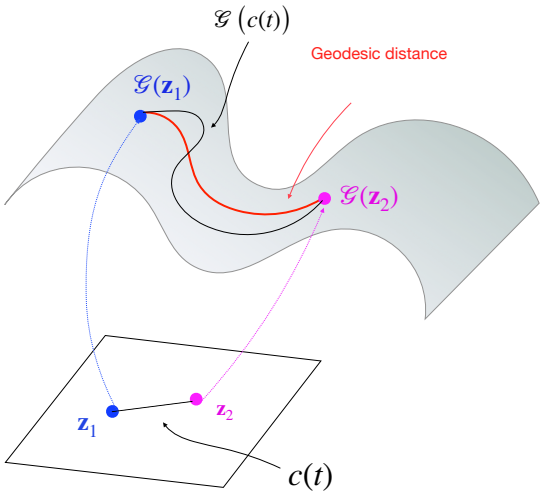


Figure 4.2. Illustration of the distance penalty. The length of the curve connecting the images corresponding to $\mathbf{z}_1$ and $\mathbf{z}_2$ depends on the Frobenius norm of the Jacobian of the mapping $\mathcal{G}$ as well as the Euclidean distance $\|\mathbf{z}_1 - \mathbf{z}_2\|^2$. We are interested in learning a mapping that preserves distances; we would like nearby points in the latent space to map to similar images. We hence use the norm of the Jacobian as the regularization prior, with the goal of preserving distances.

4.3.3 Latent vector regularization penalty

The time frames in a dynamic time series have extensive redundancy between adjacent frames, which is usually capitalized by temporal gradient regularization. Directly penalizing the temporal gradient norm of the images requires the computation of the entire image time series, which is difficult when the entire image time series is not optimized in every batch.

We consider the norm of the finite differences between images specified by $\|\nabla_p \mathbf{G}[\mathbf{z}_p]\|^2$. Using Taylor series expansion, we obtain $\nabla_p \mathbf{G}[\mathbf{z}_p] = J_{\mathbf{z}}(\mathcal{G}[\mathbf{z}])\nabla_p \mathbf{z} + \mathcal{O}(p)$. We thus have

$$\|\nabla_p \mathcal{G}[\mathbf{z}_p]\| \approx \|J_{\mathbf{z}}(\mathcal{G}[\mathbf{z}])\nabla_p \mathbf{z}\| \leq \|J_{\mathbf{z}}(\mathcal{G}[\mathbf{z}])\| \ \|\nabla_p \mathbf{z}\|. \tag{4.12}$$

Since $J_{\mathbf{z}}(\mathcal{G}[\mathbf{z}])$ is small because of the distance regularization, we propose to add a temporal regularizer on the latent vectors. For example, when applied to free-breathing cardiac MRI, we expect the latent vectors to capture the two main contributors of motion: cardiac motion and respiratory motion. The temporal regularization encourages the cardiac and respiratory phases change slowly in time.

4.3.4 Proposed optimization criterion

Based on the above analysis, we derive the parameters of the network θ and the low-dimensional latent vectors $\mathbf{z}_i; i = 1, .., M$ from the measured data by minimizing:

$$\begin{aligned} \mathcal{C}(\mathbf{z}, \theta) \ = \ & \underbrace{\sum_{i=1}^{N} \|\mathcal{A}_i \left(\mathcal{G}_\theta[\mathbf{z}_i]\right) - \mathbf{b}\|^2}_{\text{data term}} + \lambda_1 \quad \underbrace{\|J_{\mathbf{z}} \mathcal{G}_\theta(\mathbf{z})\|^2}_{\text{distance regularization}} \\ & + \lambda_2 \quad \underbrace{\|\nabla_t \mathbf{z}_t\|^2}_{\text{latent regularization}} \end{aligned} \tag{4.13}$$

with respect to $\mathbf{z}$ and θ. We use the ADAM optimization to determine the optimal parameters, and random initialization is used for the network parameters and latent variables.

A potential challenge with directly solving (4.13) is its high computational complexity. Unlike supervised neural network approaches that offer fast inference, the proposed approach optimizes the network parameters based on the measured data. This strategy will amount to a long reconstruction time when there are several image frames in the time series.

4.3.5 Strategies to reduce computational complexity

To minimize the computational complexity, we now introduce some approximation strategies.

4.3.5.1 Approximate data term for accelerated convergence

When the data is measured using non-Cartesian sampling schemes, M non-uniform fast Fourier transform (NUFFT) evaluations are needed for the evaluation of the data term, where M is the number of frames in the dataset. Similarly, M inverse non-uniform fast Fourier transform (INUFFT) evaluations are needed for each back-propagation step. These NUFFT evaluations are computationally expensive, resulting in slow algorithms. In addition, most non-Cartesian imaging schemes over-sample the center of k-space. Since the least-square loss function in (4.5) weights errors in the center of k-space higher than in outer k-space regions, it is associated with slow convergence.

To speed up the intermediate computations, we propose to use gridding with

density compensation, together with a projection step for the initial iterations. Specifically, we will use the approximate data term

$$D(\mathbf{z},\theta)=\sum_{i=1}^{M}\left\|\mathcal{P}_i\left(\mathcal{G}_\theta[\mathbf{z}_i]\right)-\mathbf{g}_i\right\|^2 \tag{4.14}$$

instead of $\sum_i \left\|\mathcal{A}_i\left(\mathcal{G}[\mathbf{z}_i]\right)-\mathbf{b}_i\right\|^2$ in early iterations to speed up the computations. Here, $\mathbf{g}_i$ are the gridding reconstructions

$$\mathbf{g}_i=\left(\mathcal{A}_i^H\mathcal{A}_i\right)^\dagger\mathcal{A}_i^H\ \mathbf{b}_i\approx\mathcal{A}_i^H\ \mathcal{W}\ \mathbf{b}, \tag{4.15}$$

where, $\mathcal{W}$ are diagonal matrices corresponding to multiplication by density compensation factors. The operators $\mathcal{P}_i$ in (4.14) are projection operators:

$$\mathcal{P}_i\ \mathbf{x}=\left(\mathcal{A}_i^H\mathcal{A}_i\right)^\dagger\left(\mathcal{A}_i^H\mathcal{A}_i\right)\ \mathbf{x}\approx\left(\mathcal{A}_i^H\ \mathcal{W}\ \mathcal{A}_i\right)\mathbf{x} \tag{4.16}$$

We note that the term $\left(\mathcal{A}_i^H\ \mathcal{W}\ \mathcal{A}_i\right)\mathbf{x}$ can be efficiently computed using Toeplitz embedding, which eliminates the need for expensive NUFFT and INUFFT steps. In addition, the use of the density compensation serves as a preconditioner, resulting in faster convergence. Once the algorithm has approximately converged, we switch the loss term back to (4.5) since it is optimal in a maximum likelihood perspective.

4.3.5.2 Progressive training-in-time

To further speed up the algorithm, we introduce a progressive training strategy, which is similar to multi-resolution strategies used in image processing. In particular, we start with a single frame obtained by pooling the measured data from all the time frames. Since this average frame is well-sampled, the algorithm promptly converges to the optimal solution. The corresponding network serves as a good initialization

for the next step. Following convergence, we increase the number of frames. The optimal θ parameters from the previous step are used to initialize the generator, while the latent vector is initialized by the interpolated version of the latent vector at the previous step. This process is repeated until the desired number of frames is reached.

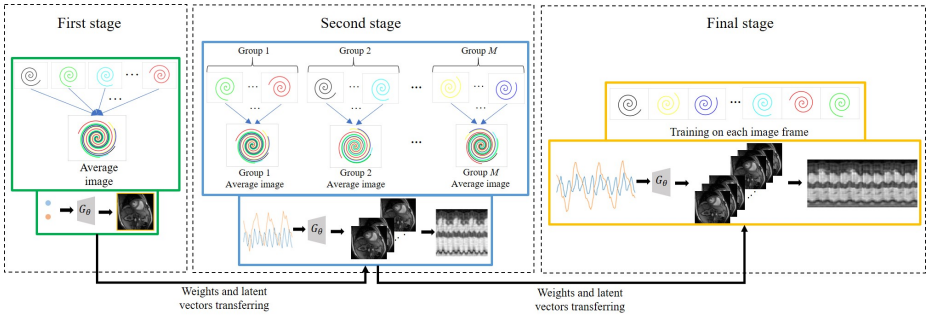


Figure 4.3. Illustration of the progressive training-in-time approach. In the first level of training, the k-space data of all the frames are binned into one and we try to solve for the average image in this level. Upon the convergence of the first step, the parameters and latent variables are transferred as the initialization of the second step. In the second level of training, we divide the k-space data into M groups and try to reconstruct the M average images. Following the convergence, we can move to the final level of training, where the parameters obtained in the second step and the linear interpolation of the latent vectors in the second step are chosen as the initializations of the final step of training.

This progressive training-in-time approach significantly reduces the computational complexity of the proposed algorithm. In this work, we used a three-step algorithm. However, the number of steps (levels) of training can be chosen based on the dataset. This progressive training-in-time approach is illustrated in Fig. 4.3.

4.4 Implementation details and datasets

4.4.1 Structure of the generator

The structure of the generator used in this work is given in Table. 4.1. The output images have two channels, which correspond to the real and imaginary parts of the MR images. Note that we have a parameter d in the network. This user-defined parameter controls the size of the generator or, in other words, the number of trainable parameters in the generator. We also have a number $\ell(\mathbf{z})$ as a user-defined parameter. This parameter represents the number of elements in each latent vector. In this work, it is chosen as $\ell(\mathbf{z}) = 2$ as we have two motion patterns in cardiac images. We use leaky ReLU for all the non-linear activations, except at the output layer, where it is tanh activation.

Input size	filter sz	# filters	Padding	Stride	Output size
$1 \times 1 \times \ell(\mathbf{z})$	1×1	100	0	1	$1 \times 1 \times 100$
$1 \times 1 \times 100$	3×3	$8d$	0	1	$3 \times 3 \times 8d$
$3 \times 3 \times 8d$	3×3	$8d$	0	1	$5 \times 5 \times 8d$
$5 \times 5 \times 8d$	4×4	$4d$	1	2	$10 \times 10 \times 4d$
$10 \times 10 \times 4d$	4×4	$4d$	1	2	$20 \times 20 \times 4d$
$20 \times 20 \times 4d$	3×3	$4d$	0	2	$41 \times 41 \times 4d$
$41 \times 41 \times 4d$	5×5	$2d$	1	2	$85 \times 85 \times 2d$
$85 \times 85 \times 2d$	4×4	d	1	2	$170 \times 170 \times d$
$170 \times 170 \times d$	4×4	d	1	2	$340 \times 340 \times d$
$340 \times 340 \times d$	3×3	2	1	2	$340 \times 340 \times 2$

Table 4.1. Architecture of the generator $\mathcal{G}_\theta$. $\ell(\mathbf{z})$ means the number of elements in each latent vector.

4.4.2 Datasets

This research study was conducted using data acquired from human subjects. The Institutional Review Board at the local institution (The University of Iowa) approved the acquisition of the data, and written consents were obtained from all subjects. The experiments reported in this chapter are based on datasets collected in the free-breathing mode using the golden angle spiral trajectory.We acquired eight datasets on a GE 3T scanner. One dataset was used to identify the optimal hyper-parameters of all the algorithms in the proposed scheme. We then used the hyperparameters to generate the experimental results for all the remaining datasets reported in this chapter. The sequence parameters for the datasets are: TR = 8.4 ms, FOV = 320 mm× 320 mm, flip angle = 18°, slice thickness = 8 mm. The datasets were acquired using a cardiac multichannel array with 34 channels. We used an automatic algorithm to pre-select the eight best coils, that provide the best signal to noise ratio in the region of interest. The removal of the coils with low sensitivities provided improved reconstructions [151]. We used a PCA-based coil combination using SVD such that the approximation error $< 5\%$. We then estimated the coil sensitivity maps based on these virtual channels using the method of Walsh et al. [135] and assumed they were constant over time.

For each dataset in this research, we binned the data from six spiral interleaves corresponding to 50 ms temporal resolution. If a Cartesian acquisition scheme with $TR = 3.5ms$ were used, this would correspond to ≈14 lines/frame; with a 340×340 matrix, this corresponds roughly to an acceleration factor of 24. Moreover, each

dataset has more than 500 frames. During reconstruction, we omit the first 20 frames in each dataset and use the next 500 frames for SToRM reconstructions; this is then used as the simulated ground truth for comparisons. The experiments were run on a machine with an Intel Xeon CPU at 2.40 GHz and a Tesla P100-PCIE 16GB GPU.

4.4.3 Quality evaluation metric

In this work, the quantitative comparisons are made using the Signal-to-Error Ratio (SER) metric (in addition to the standard Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index Measure (SSIM)) defined as:

$$\text{SER} = 20 \cdot \log_{10} \frac{\|\mathbf{x}_{orig}\|}{\|\mathbf{x}_{orig} - \mathbf{x}_{recon}\|}.$$

Here $\mathbf{x}_{orig}$ and $\mathbf{x}_{recon}$ represent the ground truth and the reconstructed image. The unit for SER is decibel (dB).

The SER metric requires a reference image, which is chosen as the SToRM reconstruction with 500 frames. However, we note that this reference may be imperfect and may suffer from blurring and related artifacts. Hence, we consider the Blind/referenceless Image Spatial Quality Evaluator (BRISQUE) [74] to evaluate the score of the image quality. The BRISQUE score is a perceptual score based on the support vector regression model trained on an image database with corresponding differential mean opinion score values. The training image dataset contains images with different distortions. A smaller score indicates better perceptual quality.

4.4.4 State-of-the-art methods for comparison

We compare the proposed scheme with the recent state-of-the-art methods for free-breathing and ungated cardiac MRI. We note that while there are many deep learning algorithms for static MRI, those methods are not readily applicable to our setting.

- Analysis SToRM [5, 100], published in 2020: The manifold Laplacian matrix is estimated from k-space navigators using kernel low-rank regularization, followed by solving for the images using (4.4).

- Time-DIP [54] implementation based on the arXiv form at the submission of this article: This is an unsupervised learning scheme, that fixes the latent variables as noise and solves for the generator parameters. For real-time applications, Time-DIP chooses a preset period, and the noise vectors of the frames corresponding to the multiples of the period were chosen as independent Gaussian variables [54]. The latent variables of the intermediate frames were obtained using linear interpolation. We chose a period of 20 frames, which roughly corresponds to the period of the heart beats.

- Low-rank [66]: The image frames in the time series are recovered using the nuclear norm minimization.

4.4.5 Hyperparameter tuning

We used one of the acquired datasets to identify the hyperparameters of the proposed scheme. Since we do not have access to the fully-sampled dataset, we used

the SToRM reconstructions from 500 images (acquisition time of 25 seconds) as a reference. The smoothness parameter λ of this method was manually selected as $\lambda = 0.01$ to obtain the best recovery, as in the literature [5]. All of the comparisons relied on image recovery from 150 frames (acquisition time of 7.5 seconds). The hyperparameter tuning approach yielded the parameters $d = 40$, $\lambda_1 = 0.0005$, and $\lambda_2 = 2$ for the proposed approach. We demonstrate the impact of tuning d in Fig. 4.6, while the impact of choosing λ_1 and λ_2 is shown in Fig. 4.4. The hyperparameter optimization of SToRM from 150 frames resulted in the optimal smoothness parameter $\lambda = 0.0075$. For Time-DIP, we follow the design of the network shown by Jin et al. [54], where the generator consists of multiple layers of convolution and upsampling operations. To ensure fair comparison, we used a similar architecture, where the base size of the network was tuned to obtain the best results.

We use a three-step progressive training strategy. In the first step, the learning rate for the network is 1×10^{-3} and 1000 epoches are used. For the second step of training, the learning rate for the network is 5×10^{-4} and the learning rate for the latent variable is 5×10^{-3}. In this stage, 600 epoches are used. In the final step of training, the learning rate for the network is 5×10^{-4}, the learning rate for the latent variable is 1×10^{-3}, and 700 epoches are used.

4.5 Experiments and results

4.5.1 Impact of different regularization terms

We first study the impact of the two regularization terms in (4.13). The parameter d corresponding to the size of the network (see Table 4.1) was chosen as $d = 24$

in this case. In Fig. 4.4 (a), we plot the reconstruction performance with respect to the number of epoches for three scenarios: (1) using both regularization terms; (2) using only latent regularization; and (3) using only distance/network regularization. In the experiment, we use 500 frames of SToRM ($\sim$ 25 seconds of acquisition) reconstructions, which is called "SToRM500", as the reference for SER computations. We tested the reconstruction performance for the three scenarios using 150 frames, which corresponds to around 7.5 seconds of acquisition. From the plot, we observe that without using the network regularization, the SER degrades with increasing epoches, which is similar to that of DIP. In this case, an early stopping strategy is needed to obtain good recovery. The latent vectors corresponding to this setting are shown in (c), which shows mixing between cardiac and respiratory waveforms. When latent regularization is not used, we observe that the SER plot is roughly flat, but the latent variables show quite significant mixing, which translates to blurred reconstructions. By contrast, when both network and latent regularizations are used, the algorithm converges to a better solution. We also note that the latent variables are well decoupled; the blue curve captures the respiratory motion, while the orange one captures the cardiac motion. We also observe that the reconstructions agree well with the SToRM reconstructions. The network now learns meaningful mappings, which translate to improved reconstructions when compared to the reconstructions obtained without using the regularizers.

4.5.2 Benefit of progressive training-in-time approach

In Fig. 4.5, we demonstrate the significant reduction in run-time offered by the progressive training strategy described in Section 4.3.5.2. Here, we consider the recovery from 150 frames with and without the progressive strategy. Both regularization priors were used in this strategy, and d was chosen as 24. We plot the reconstruction performance, measured by the SER with respect to the running time. The SER plots show that the proposed scheme converges in around ≈ 200 seconds, while the direct approach takes more than 2000 seconds. We also note from the SER plots that the solution obtained using progressive training is superior to the one without progressive training.

4.5.3 Impact of size of the network

The architecture of the generator $\mathcal{G}_\theta$ is given in Table 4.1. Note that the size of the network is controlled by the user-defined parameter d, which dictates the number of convolution filters and hence the number of trainable parameters in the network. In this section, we investigate the impact of the user-defined parameter d on the reconstruction performance. We tested the reconstruction performance using $d = 8, 16, 24, 32, 40,$ and 48, and the obtained results are shown in Fig. 4.6. From the figure, we see that when $d = 8$ or $d = 16$, the generator network is too small to capture the dynamic variations. When $d = 8$, the generator is unable to capture both cardiac motion and respiratory motion. When $d = 16$, part of the respiratory motion is recovered, while the cardiac motion is still lost. The best SER scores with respect to SToRM with 500 frames is obtained for $d = 24$, while the lowest Brisque

scores are obtained for $d = 40$. We also observe that the features including papillary muscles and myocardium in the $d = 40$ results appear sharper than those of SToRM with 500 frames, even though the proposed reconstructions were only performed from 150 frames. We use $d = 40$ for the subsequent comparisons in the chapter.

4.5.4 Comparison with the state-of-the-art methods

In this section, we compare our proposed scheme with several state-of-the-art methods for the reconstruction of dynamic images.

Methods	SToRM500	SToRM150	Propsed	Time-DIP
SER (dB)	NA	17.3	**18.2**	16.7
PSNR (dB)	NA	32.7	**33.5**	32.0
SSIM	NA	0.86	**0.89**	0.87
Brisque	**35.2**	40.2	37.1	42.9
Time (min)	47	13	17	57

Table 4.2. Quantitative comparisons based on six datasets: We used six datasets to obtain the average SER, PSNR, SSIM, Brisque score, and time used for reconstruction.

In Fig. 4.7, we compare the region of interest for SToRM500, SToRM with 150 frames (SToRM150), the proposed method with two different d values, the unsupervised Time-DIP approach, and the low-rank algorithm. From Fig. 4.7, we observe that the proposed scheme can significantly reduce errors in comparison to SToRM150. Additionally, the proposed scheme is able to capture the motion patterns better than Time-DIP, while the low-rank method is unable to capture the motion patterns. From the time profile in Fig. 4.7, we notice that the proposed scheme is capable of recov-

ering the abrupt change in blood-pool contrast between diastole and systole. This is due to inflow effects associated with gradient echo (GRE) acquisitions. In particular, the blood from regions outside the slice enters the heart, which did not experience any of the former slice-selective excitation pulses; the differences in magnetization of the blood with no magnetization history, and that was within the slice, results in the abrupt change in intensity. We note that some of the competing methods such as Time-DIP and low-rank, blur these details.

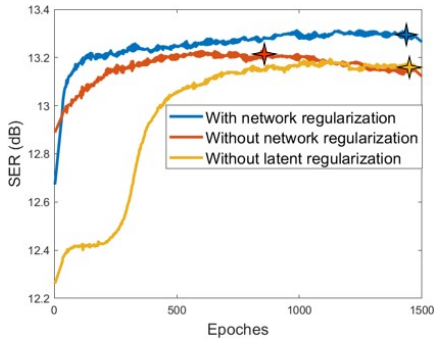
We also perform the comparisons on a different dataset in Fig. 4.8. We compare the proposed scheme with SToRM500, SToRM150, Time-DIP, and the low-rank approach. The results are shown in Fig. 4.8. From the figure, we see that the proposed reconstructions appear less blurred than those of the conventional schemes.

We also compared the proposed scheme with SToRM500, SToRM150, and the unsupervised Time-DIP approach quantitatively. We omit the low-rank method here because low-rank approach often failed in some datasets. The quantitative comparisons are shown in Table 4.2. We used SToRM500 as the reference for SER, PSNR, and SSIM calculations. The quantitative results are based on the average performance from six datasets.

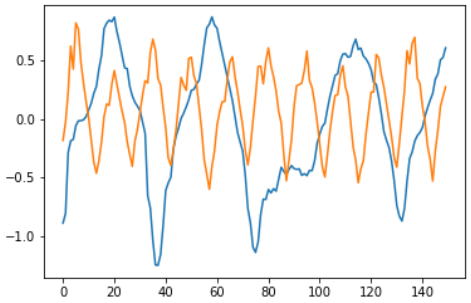
Finally, we illustrate the proposed approaches in Fig. 4.9 and Fig. 4.10, respectively. The proposed approach decoupled the latent vectors corresponding to the cardiac and respiratory phases well, as shown in the representative examples in Fig. 4.9 (a) and Fig. 4.10 (a).

4.6 Conclusion

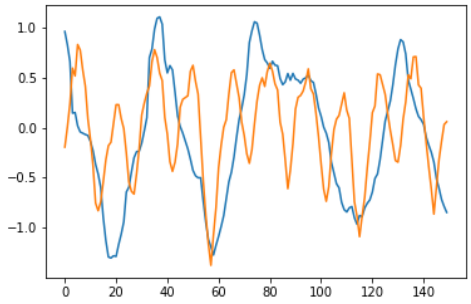
In this work, we introduced an unsupervised generative SToRM framework for the recovery of free-breathing cardiac images from spiral acquisitions. This work assumes that the images are generated by a non-linear CNN-based generator $\mathcal{G}_\theta$, which maps the low-dimensional latent variables to high-resolution images. Unlike traditional supervised CNN methods, the proposed approach does not require any training data. The parameters of the generator and the latent variables are directly estimated from the undersampled data. The key benefit for this generative model is its ability to compress the data, which results in a memory-effective algorithm. To improve the performance, we introduced a network/distance regularization and a latent variable regularization. The combination of the priors ensures the learning of representations that preserve distances and ensure the temporal smoothness of the recovered images; the regularized approach provides improved reconstructions while minimizing the need for early stopping. To reduce the computational complexity, we introduced a fast approximation of the data loss term as well as a progressive training-in-time strategy. These approximations result in an algorithm with computational complexity comparable to our prior SToRM algorithm. The main benefits of this scheme are the improved performance and considerably reduced memory demand. While our main focus in this work was to establish the benefits of this work in 2D, we plan to extend this work to 3D applications in the future.



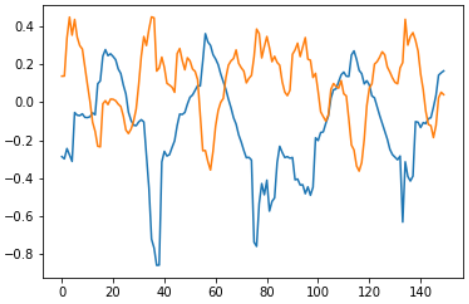
(a) Performance comparison



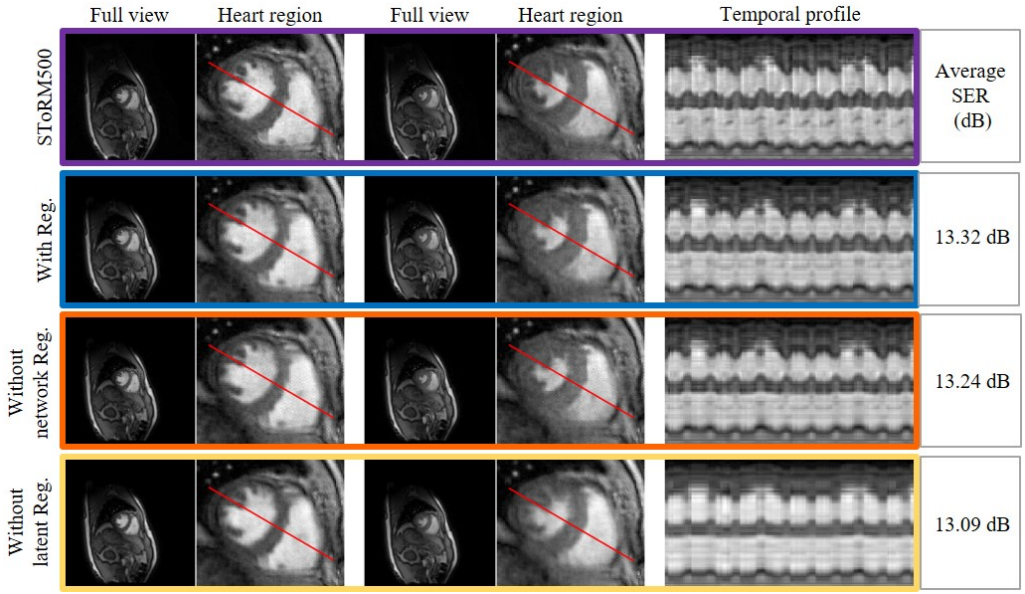
(b) Latent codes with both terms



(c) Without distance regularization



(d) Without latent regularization



(e) Visual and quantitative comparisons

Figure 4.4. Illustration of the impact of the regularization terms in the proposed scheme with $d = 24$. We considered three cases in the experiment: (1) using both regularizations, (2) using only latent regularization, and (3) using only network regularization; these correspond to the blue, orange, and yellow curves in (a). In (b), (c), and (d), we showed the learned latent vectors for the three cases. The visual and quantitative comparisons of the three cases are shown in (e).

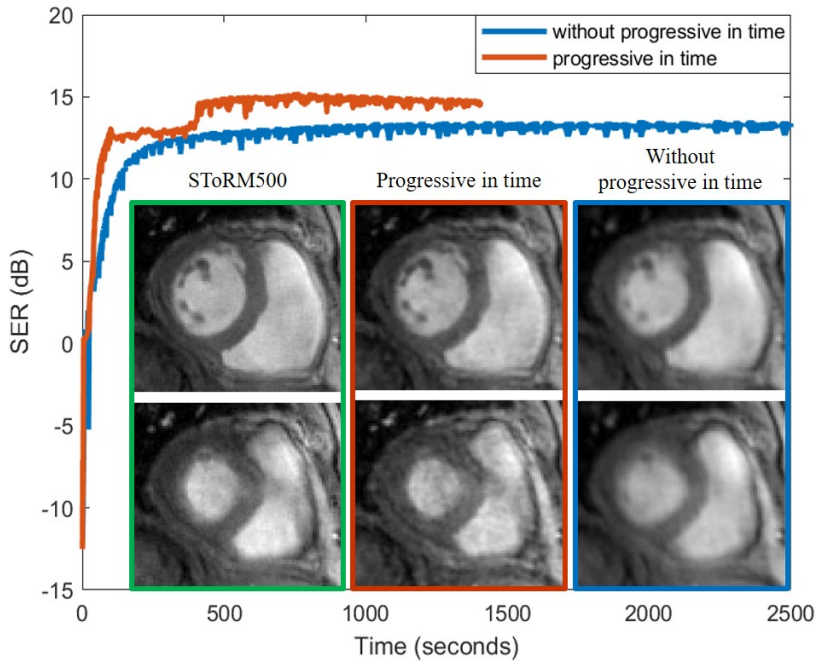


Figure 4.5. Comparisons of the reconstruction performance with and without the progressive training-in-time strategy using $d = 40$. From the plot of SER vs. running time, we can see that the progressive training-in-time approach yields better results with much less running time comparing to the training without using progressive training-in-time. Two reconstructed frames near the end of systole and diastole using SToRM500, the proposed scheme with progressive training-in-time and the proposed scheme without using the progressive training-in-time are shown in the plot as well for comparison purposes. The average Brisque scores for SToRM500, the reconstruction with progressive training-in-time, and the reconstruction without progressive training-in-time are $36.4, 37.3$ and 39.1 respectively.

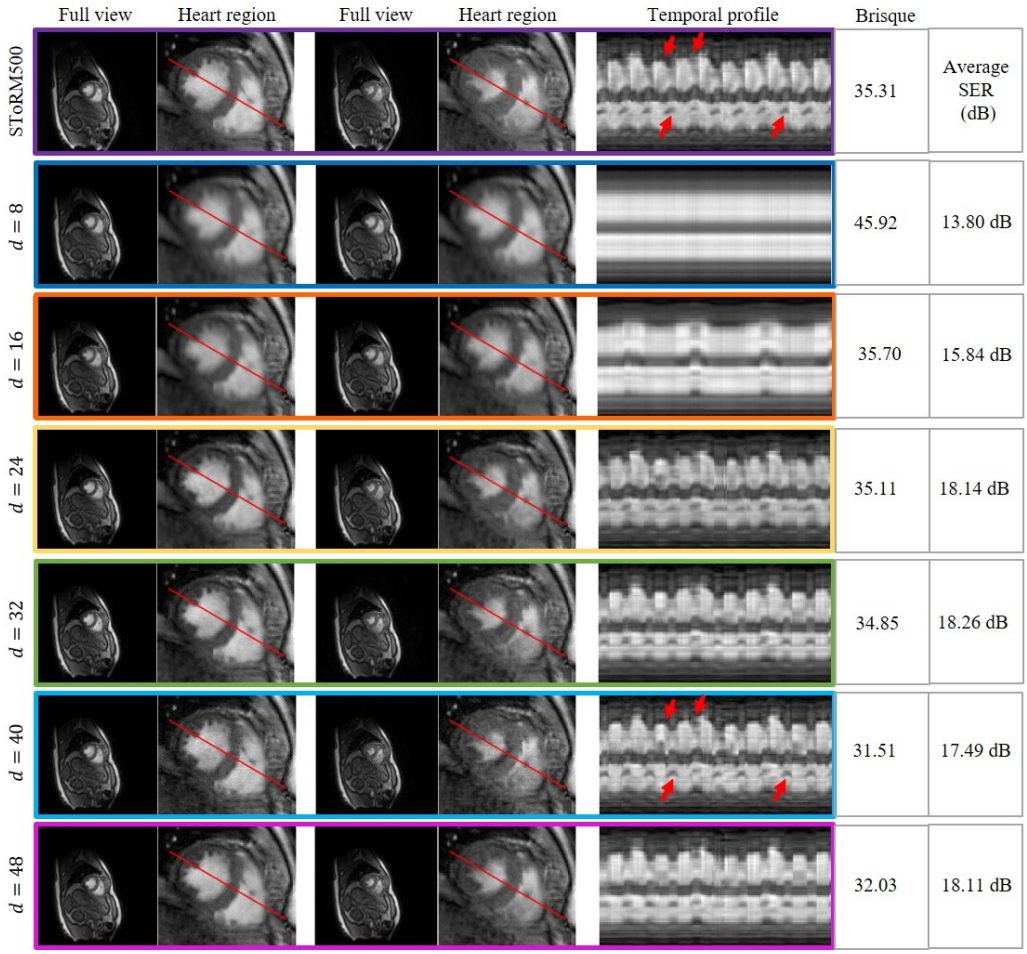
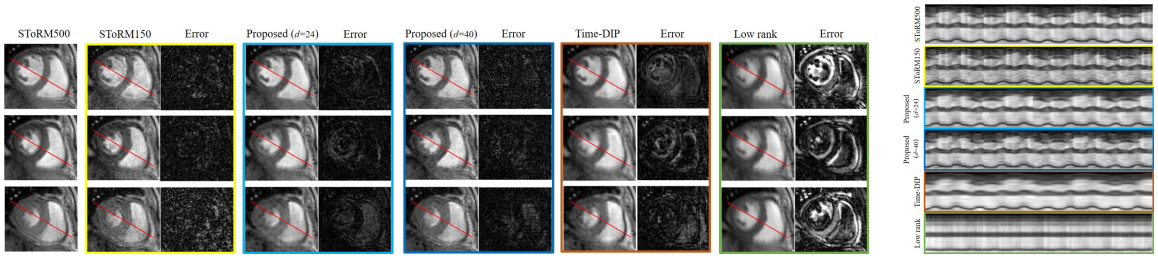
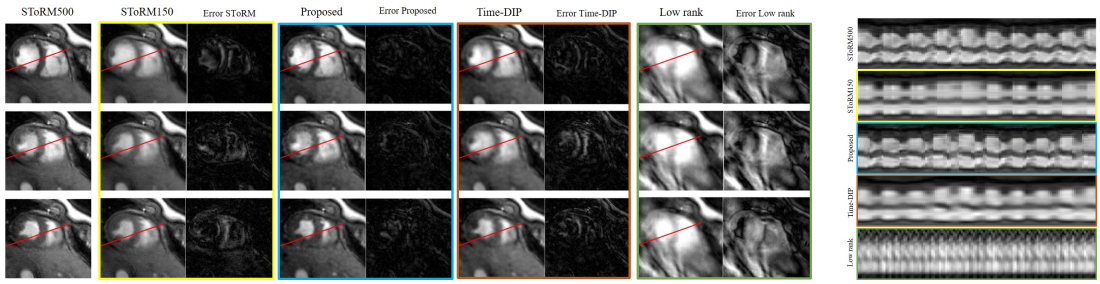


Figure 4.6. Impact of network size on reconstruction performance. In the experiments, we chose $d = 8, 16, 24, 32, 40$ and 48 to investigate the reconstruction performance. We used 500 frames for SToRM reconstructions (SToRM500) as the reference for SER comparisons. For the investigation of the impact of network size on the reconstructions, we used 150 frames. The diastolic and systolic states and the temporal profiles are shown in the figure for each case. The Brisque scores and average SER are also reported. It is worth noting that when $d = 40$, the results are even less blurred than the SToRM500 results, even though only one-third of the data are used.



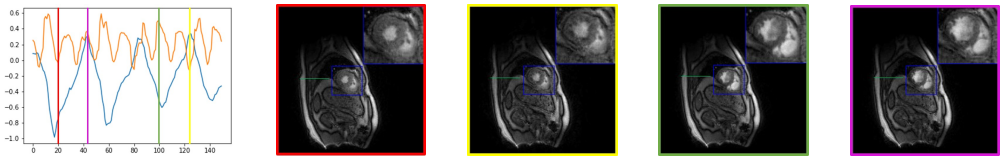
(a) Visual comparisons (b) Time profiles

Figure 4.7. Comparisons with the state-of-the-art methods. The first column of (a) corresponds to the reconstructions from 500 frames ($\sim$ 25s of acquisition time), while the rest of the columns are recovered from 150 frames ($\sim$ 7.5s of acquisition time). The top row of (a) corresponds to the diastole phase, while the third row is the diastole phase. The second row of (a) is an intermediate one. Fig. (b) corresponds to the time profiles of the reconstructions. We observe that the proposed ($d = 40$) reconstructions exhibit less blurring and fewer artifacts when compared to SToRM150 and competing methods.



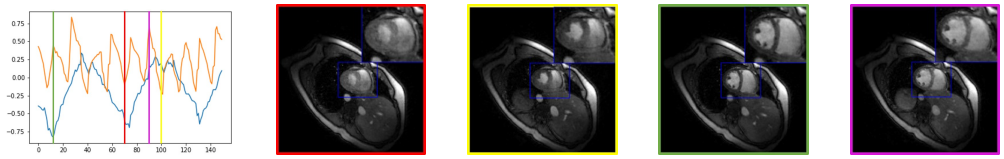
(a) Visual comparisons (b) Time profiles

Figure 4.8. Comparisons with the state-of-the-art methods. The first column of (a) corresponds to the reconstructions from 500 frames ($\sim$ 25s of acquisition time), while the rest of the columns are recovered from 150 frames ($\sim$ 7.5s of acquisition time). The top row of (a) corresponds to the diastole phase, while the third row is the diastole phase. The second row of (a) is an intermediate one. Fig. (b) corresponds to the time profiles of the reconstructions. We chose $d = 40$ for the proposed scheme. We observe that the proposed reconstructions appear less blurred when compared to the conventional schemes.



(a) Latent vectors (b) Systole in E-E (c) Systole in E-I (d) Diastole in E-E (e) Diastole in E-I

Figure 4.9. Illustration of the framework of the proposed scheme with $d = 40$. We plot the latent variables of 150 frames in a time series on the first dataset. We showed four different phases in the time series: systole in End-Expiration (E-E), systole in End-Inspiration (E-I), diastole in End-Expiration (E-E), and diastole in End-Inspiration (E-I). A thin green line surrounds the liver in the image frame to indicate the respiratory phase. The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors.



(a) Latent vectors (b) Systole in E-E (c) Systole in E-I (d) Diastole in E-E (e) Diastole in E-I

Figure 4.10. Illustration of the framework of the proposed scheme with $d = 40$. We plot the latent variables of 150 frames in a time series. We showed four different phases in the time series: systole in End-Expiration (E-E), systole in End-Inspiration (E-I), diastole in End-Expiration (E-E), and diastole in End-Inspiration (E-I). The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors.

CHAPTER 5

ALIGNED & JOINTLY RECOVERY OF MULTI-SLICE DATA USING UNION OF SURFACES PRIOR

5.1 Introduction

Breath-held cine imaging, which provides valuable indicators of abnormal structure and function, is an integral part of cardiac MRI exams. Multi-slice protocols, which offer good in-flow contrast between the myocardium and the blood, are often preferred over 3D acquisitions. A challenge with multi-slice approaches is the potential for mismatches between slices resulting from inconsistent breath-holds. Another challenge is the difficulty in acquiring data from subjects who cannot comply with multiple long breath-holds. Compressed sensing methods have been widely used to reduce the breath-hold duration [8, 57, 60, 69]. Recently, deep learning methods have emerged as powerful options to accelerate cardiac cine MRI, with excellent performance [20, 61, 104, 112, 141]. Despite these advances, several subject groups, such as pediatric and chronic obstructive pulmonary disease (COPD) subjects, cannot comply with breath-held acquisitions.

Several authors have introduced self-gating and manifold methods for free-breathing and ungated imaging applications. Self-gating methods [24, 32, 40, 41, 108, 152] use k-space navigators to estimate the cardiac/respiratory phase, followed by the binning and recovery of binned images. Manifold approaches [22, 80, 81, 118, 132], including the smoothness regularization on manifolds (SToRM) approach [5, 97, 100], which perform soft-gating based on k-space navigators, are emerging as alternatives

to self-gating. All of the above schemes perform the independent recovery of multi-slice data; they fail to capitalize on the extensive redundancies between nearby slices. Moreover, current manifold approaches, which recover all the image frames in the time series, are associated with high memory demand. In addition, sophisticated post-processing approaches are often needed to temporally align the data from different slices [5, 22].

In this note, we introduce a deep generative model for the joint reconstruction of multi-slice MRI, termed as generative multi-slice SToRM (g-SToRM:MS). This is the multi-slice generalization of the recent generative single-slice SToRM (g-SToRM:SS) framework [154] that has conceptual similarities with time dependent deep image prior [56]. The g-SToRM:SS algorithm models the images in the time-series from each slice as a smooth, non-linear function of a low-dimensional latent vector. The patient-specific non-linear function is represented as a convolutional neural network (CNN), which is the same for all the time frames. By contrast, the latent vectors capture the temporal variability in the data, including cardiac and respiratory motion. This approach exploits the structural bias of CNN to images to offer implicit regularization [63], thus improving the results compared to classical analysis single-slice SToRM (a-SToRM:SS) [5, 100]. Unlike current CNN-based approaches, g-SToRM:SS does not require the fully sampled training data, which is not available in the free-breathing setting. The patient-specific CNN parameters and the latent vectors are learned from only the highly under-sampled k-t space measurements of the subject. Moreover, unlike prior methods [5, 24, 80, 81, 97, 100, 118], g-SToRM:SS

does not require k-space navigators to estimate the motion patterns.

The g-SToRM:SS [154] and a-SToRM:SS [5] algorithms, as well as current single-slice methods [24,80,81,97,100,118], perform the independent reconstruction of the slices. These approaches are not capable of exploiting the redundancies between adjacent slices. To exploit the extensive redundancies between adjacent slices, we propose to use a 3D generator to model the volume corresponding to all the slices at each time point. We use a 2D spiral gradient echo sequence (GRE) to acquire the data from each slice. We use the same 3D generator for all the time frames and slices. We propose to use different latent vectors for each slice to account for the differences between cardiac and respiratory motion during the acquisition of the different slices. The latent vectors for each slice/time as well as the CNN parameters are jointly learned from the measured data of all the slices. Once the learning is complete, the generator can be excited with the latent vector of any slice to generate an aligned multi-slice volume when it generates the aligned multi-slice data with matching cardiac/respiratory phases. In addition to enabling the exploitation of the inter-slice redundancies, this approach also simplifies the image processing workflow and subsequent processing. The proposed scheme is illustrated in Fig. 5.1.

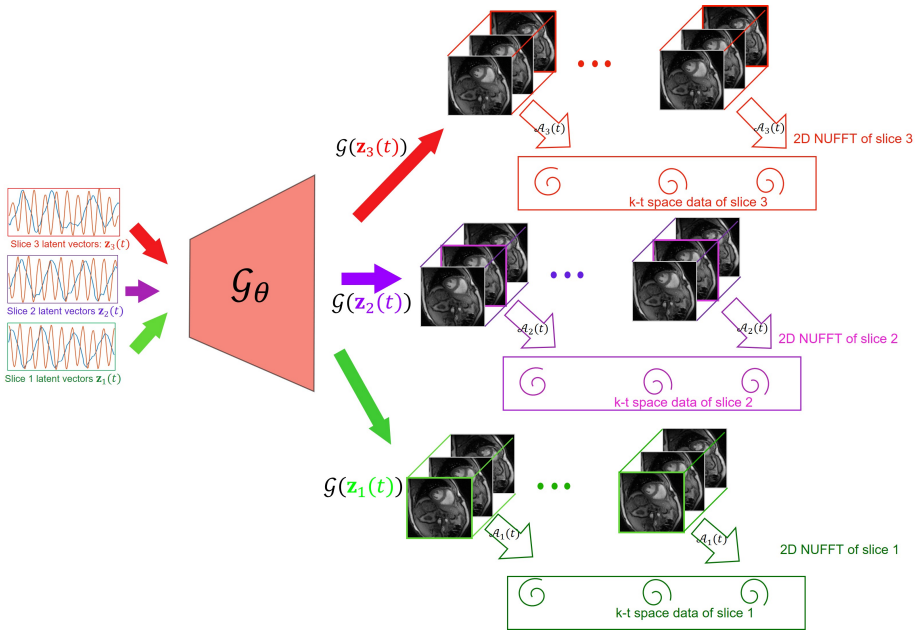


Figure 5.1. Illustration of the proposed scheme on a dataset with three slices. The latent vectors of the i^{th} slice and time instant t, denoted by $\mathbf{z}_{i,t}$, are fed into the deep generative model $\mathcal{G}_\theta$, which generates the multi-slice image volume $\rho_{i,t} = \mathcal{G}_\theta[\mathbf{z}_{i,t}]$. The latent vectors $\mathbf{z}_{i,t}$ and the parameters θ of the generative model are learned jointly from the entire k-t space data $\mathbf{b}_{i,t}, \forall i, t$. The data consistency term in (5.3) specified by $\sum_i \sum_t \|\mathcal{A}_{it}(\rho_{i,t}) - \mathbf{b}_{i,t}\|^2$ is the sum of the errors between the measured k-t space data of each slice and the multi-channel measurements of the corresponding slices. For example, the operator $\mathcal{A}_{it}$ extracts the i^{th} slice from $\rho_{i,t}$ and evaluates its multi-channel Fourier transform, which is compared with the measurements $\mathbf{b}_{i,t}$. We additionally use regularization priors on the network and the latent parameters to make the reconstruction problem well posed.

The g-SToRM:MS framework uses the mean square error loss function

$$\sum_{i=1}^{M} \sum_{t=1}^{N} \|\mathcal{A}_{it}\left(\mathcal{G}_\theta[\mathbf{z}_{i,t}]\right) - \mathbf{b}_{i,t}\|^2$$

for image recovery (see Fig. 5.1). In particular, we feed the latent vectors $z_{i,t}$ corresponding to the i^{th} slice and the t^{th} time frame to the generator $\mathcal{G}_\theta$, which outputs the corresponding 3D volume $\rho_{i,t} = \mathcal{G}_\theta[z_{i,t}]$. The forward model $\mathcal{A}_{i,t}$ extracts the i^{th} slice of $\rho_{i,t}$ and computes the multi-channel non-uniform Fourier transform, which is compared with the k-space data of the i^{th} slice and the t^{th} time frame. The CNN

parameters θ and the latent variables are obtained by minimizing the above cost function using ADAM [58]. We note that the norm of the gradient of the CNN, denoted by $||\nabla_{\mathbf{z}}\mathcal{G}||_F^2$, is a measure of the smoothness of the recovered images on the image manifold [154]; we add the above penalty to regularize the learning as in [154]. In addition, as we expect the image frames in the time series to vary smoothly in time, we also use a regularization term to penalize the temporal smoothness of the latent vectors as in [154]. We note that the latent vectors suffer from non-uniqueness; for every set of latent vectors and network combinations, one could come up with several other combinations. For instance, one could scale the latent vector with an arbitrary invertible matrix, while the fully connected weights of the first layer of the network can undo this scaling. This non-uniqueness is not a big concern in the single-slice setting [154]. In the multi-slice setting, this non-uniqueness can result in image quality degradation when one is using the latent vectors of one slice to generate the entire volume. Specifically, the probability distribution of the latent vectors of one slice could be drastically different from that of other slices. To minimize these issues, we use an additional Kullback-Leibler (K-L) divergence [145] penalty on the latent vectors of each slice to encourage their probability densities to be a zero mean Gaussian with the covariance matrix as identity. The memory footprint of the algorithm is determined by the size of the network as well as the latent vectors $\mathbf{z}$, which is orders of magnitude smaller than that of manifold approaches, which often require the recovery of each time frame in the time series.

5.2 Methods

5.2.1 Forward model

We consider the recovery of the 3D time-series $\rho_{i,t}(\mathbf{r})$, where $\mathbf{r} = (x, y, z)$ represents the spatial coordinates and t denotes the time from the multi-slice data. Here, z is the slice location. We model the multi-slice acquisition of the data as

$$\mathbf{b}_{i,t} = \mathcal{A}_{it}\Big(\rho_{i,t}(\mathbf{r})\Big) + \mathbf{n}_{i,t}, \tag{5.1}$$

where $\mathcal{A}_{it}$ extracts the i^{th} slice of the volume $\rho(\mathbf{r})$ at time point t and evaluates the multi-channel Fourier measurements on the trajectory $k_{i,t}$ corresponding to the time point t. $\mathbf{n}_{i,t}$ represents the noise. The main problem we consider is to recover the volume time series $\rho(\mathbf{r})$ from the noisy under-sampled measurements $\mathbf{b}_{i,t}$.

5.2.2 Proposed approach

In this work, we model the image volume at the time point t during the acquisition of the i^{th} slice, denoted by $\rho_{i,t}(\mathbf{r})$, as the non-linear mapping:

$$\rho_{i,t}(\mathbf{r}) = \mathcal{G}_\theta\left[\mathbf{z}_{i,t}\right] \tag{5.2}$$

Here, $\mathbf{z}_{i,t}$ are the low dimensional latent vectors corresponding to slice i at a specific time point t, while $\mathcal{G}_\theta$ is a deep CNN generator whose weights are denoted by θ. Our experiments show that 2-4 latent vectors are often sufficient to represent the data. (5.2) indicates that the images live in the range space of the non-linear mapping $\mathcal{G}_\theta$, where the domain is a low-dimensional subspace. We note that we are essentially modeling the images in the time series as points on a surface or manifold, denoted by $\mathcal{M}$. Thus, we term (5.2) as the generative SToRM model. The low-dimensional

nature of the latent vectors enables the exploitation of the non-local redundancies between images at different time points, thus facilitating the fusion of information between them as in [5, 100]. We note that the network is shared across all slices and time points; this approach facilitates the exploitation of the spatial redundancies between the slices and time points, in addition to being memory efficient. Moreover, CNNs often have a structural bias towards natural images [63], which offers implicit spatial regularization.

We propose to jointly estimate the network parameters θ and the latent variables $\mathbf{z}$ of different slices from the measured multi-slice data by minimizing the following cost function:

$$\mathcal{C}(\mathbf{z},\theta)=\sum_{i=1}^{M}\sum_{t=1}^{N}\left\|\mathcal{A}_{it}\left(\mathcal{G}_{\theta}[\mathbf{z}_{i,t}]\right)-\mathbf{b}_{i,t}\right\|^{2}+\lambda_{1}\underbrace{\left\|\nabla_{\mathbf{z}}\mathcal{G}_{\theta}\right\|^{2}}_{\text{network regularization}}+\boldsymbol{\lambda}_{2}\underbrace{\mathcal{R}(\mathbf{z})}_{\text{latent regularization}}. \tag{5.3}$$

As shown in [154], the smoothness of the image manifold is dependent upon the norm of the gradient of the CNN $\mathcal{G}_{\theta}$. Motivated by the improvement in performance resulting from the use of the manifold smoothness penalty in [154], we propose to use this term to regularize the multi-slice setting as well.

The last term in (5.3) is a regularization penalty on the latent vectors to further constrain the solution. We use a combination of the smoothness penalty and the K-L divergence penalty as latent vector priors:

$$\mathcal{R}(\mathbf{z})=\lambda_{21}\cdot||\nabla_{t}\mathbf{z}||+\lambda_{22}\cdot\mathrm{KL}\big[q(\mathbf{z})|\mathcal{N}(\mathbf{0},\mathbf{I})\big)\big]. \tag{5.4}$$

The first term $||\nabla_t\mathbf{z}||$ is a temporal smoothness penalty on the latent vectors. The

image frames are known to change slowly in time. If the gradient of the CNN $\mathcal{G}_\theta$ is finite, distances between close-by points on the image manifold $\mathcal{M}$ are closely related to the distances between the latent vectors. Since the evaluation of the temporal smoothness on the images is computationally and memory intensive, we propose to directly penalize the temporal smoothness of the latent vectors. After the training process is complete, we plan to generate the volume time series by feeding the generator with the latent variables of a particular slice. If the latent variables have different probability distributions, this approach can result in degraded image quality for other slices. We hence use an additional regularization term, which is the Kullback–Leibler (KL) divergence [34] between the probability density of the latent vectors of each slice and a Gaussian distribution with zero mean and identity covariance matrix; this is the second term in (5.4). This penalty encourages the latent vectors for each slice to have the same distribution while being maximally uncorrelated. In free-breathing cardiac MRI, it is often difficult to decouple the subtle cardiac motion from the strong respiratory motion; the low correlation between the components of the latent variables ensures that the network learns mappings between meaningful latent vectors and images.

The parameters of the network and the latent vectors are jointly learned in an unsupervised fashion from the measured k-t space data. We use the ADAM optimization algorithm for the learning. The proposed approach relies only on the under-sampled k-t space data of the specific patient. Unlike current deep learning algorithms for image recovery, the proposed approach does not require extensive

amounts of fully sampled training data.

5.2.3 Acquisition scheme

The multi-slice images are acquired sequentially using a 2D (GRE) sequence with golden angle spiral readouts in the free-breathing and ungated setting. The sequence parameters for the datasets are: FOV = 320 mm × 320 mm, flip angle = 18°, slice thickness = 8 mm. The datasets were acquired using a cardiac multi-channel array with 34 channels. The institutional review board at the University of Iowa approved the acquisition of the data, and written consents were obtained from the subjects. We show the data acquired from two healthy volunteers and a patient with COPD. At the time of data acquisition, the patient with COPD was at GOLD 3 stage [2] , but was not dependent on oxygen. The datasets from the normal volunteers were acquired on a GE MR750W scanner with a 34-channel array, while the COPD subject was scanned after the scanner was upgraded to GE Premier with a 54-channel array. We used an algorithm developed in-house to pre-select the the coils that provide the best signal-to-noise ratio in the region of interest. A PCA-based coil combination scheme was used such that the approximation error was less than 5%. We then estimated the coil sensitivity maps based on these virtual channels using ESPIRiT [129] and assumed them to be constant over time.

For the datasets from normal volunteers, a total of 3,192 spirals were acquired for each slice in the normal subjects with TR=8.4 ms, which corresponds to an acquisition time of 27 seconds/slice, where every sixth spiral was acquired with the same angle; these spirals were used for self-navigation in a-SToRM:SS. We binned the data

from six spiral interleaves corresponding to 50 ms temporal resolution. The datasets from the healthy volunteers were acquired with 8 and 5 slices, respectively. For the COPD dataset, a total of 3,200 spirals were acquired with TR=8 ms for three slices, which translates to an acquisition time of 25 seconds/slice. Every fifth spiral was acquired with the same angle, which was used for self-navigation in a-SToRM:SS. We binned five interleaves per frame, resulting in a 40 ms temporal resolution. For the normal subjects, we use the k-space data from 150 frames for reconstruction and comparison, which translates to 7.5 ms/slice; we refer to the approaches as a-SToRM:150, g-SToRM:SS, and g-SToRM:MS, respectively, in the figures. For the COPD dataset, we use the k-space data of 150 frames in each slice, which correspond to 6 seconds of acquisition time per slice. The analysis SToRM reconstructions using the entire data with 500 and 600 frames for the normal subjects and COPD subjects (referred to as a-SToRM:500 and a-SToRM:600 in the figures) are used as reference.

5.2.4 Training approach

We adopt the progressive-in-time training strategy introduced in [154] to realize a computationally efficient reconstruction. In particular, we start with the recovery of an image series with a few frames, obtained by binning more spirals per frame. Because each image is oversampled and because the number of time frames is limited, the generator and the latent vectors converge quickly in this setting. Once the algorithm has converged, we interpolate the latent vectors to a finer temporal grid and re-optimize the latent vectors and generator parameters with fewer spirals/bin. The optimal generator parameters from the previous setting are used as the initialization

in this case. In this work, we consider a three-step progression: $10 \rightarrow 50 \rightarrow 150$. Specifically, we start with ten images per slice (corresponding to 90 spirals/bin for the volunteer datasets and 75 spirals/bin for the COPD dataset), followed by 50 images per slice (corresponding to 18 spirals/bin for the volunteer datasets and 15 spirals/bin for the COPD dataset), and finally 150 images per slice (corresponding to 6 spirals/bin for the volunteer datasets and 5 spirals/bin for the COPD dataset). Results in this work were generated using an Intel Xeon CPU at 2.40 GHz and a Tesla V100-PCIE 32GB GPU.

5.2.5 Comparison with state-of-the-art methods

We compare the proposed multi-slice generative manifold approach with the following existing methods.

- Analysis SToRM [5]: The a-SToRM:SS model estimates the manifold Laplacian matrix from the k-space navigators using kernel low-rank regularization, which is then used to solve for the images. We note that the analysis SToRM approach yields comparable or improved performance to state-of-the-art self-gated methods.

- Single-slice generative SToRM [154]: The g-SToRM:SS approach uses a CNN generator to generate the single-slice image series from the highly undersampled k-t space data. It performs the independent recovery of each slice and hence fails to exploit the inter-slice redundancies.

The quantitative comparisons are made using the signal-to-error ratio (SER)

defined as

$$\text{SER} = 20 \cdot \log_{10} \frac{||\mathbf{x}_{\text{ref}}||}{||\mathbf{x}_{\text{ref}} - \mathbf{x}_{\text{recon}}||}.$$

Here $\mathbf{x}_{\text{ref}}$ and $\mathbf{x}_{\text{recon}}$ represent the reference and the reconstructed images, respectively. The unit for SER is decibel (dB). Because a-SToRM:600 or a-SToRM:500 are sometimes not the perfect references, we also report the Blind Image Spatial Quality Evaluator (BRISQUE) [74], which is a reference-less perceptual measure of image quality; a smaller score indicates better quality.

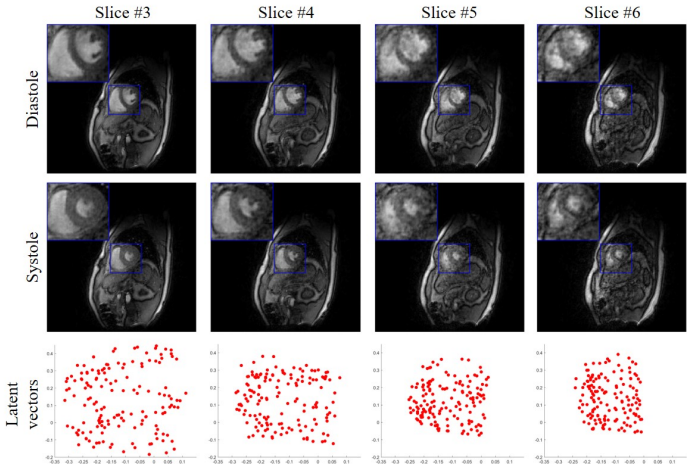
5.3 Results

The impact of the network size, λ_1 and λ_2, have been studied in [154]. We perform experiments similar to [154] to determine the best parameters on one dataset. We observe that three latent variables are sufficient in offering good reconstructions in all the cases considered in this work. Once the parameters have been identified, we use the setting for the remaining datasets.

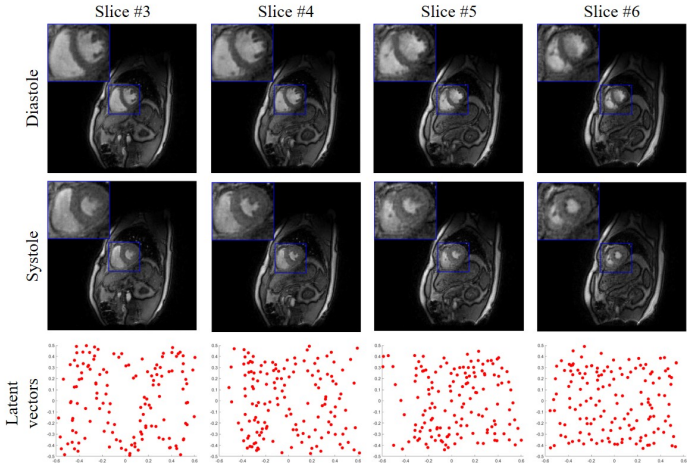
5.3.1 Impact of K-L divergence penalty

We first study the impact of the newly added K-L divergence penalty, which is expected to play a critical role in the multi-slice setting. This is important because we finally generate aligned multi-slice data with matching cardiac/respiratory phases using the latent variables of a specific slice. If the latent vectors of different slices have different probability distributions, the generator may generate image frames without any meaning for slices that differ from the chosen one. We study the impact of the K-L divergence penalty in Fig. 5.2 using four slices. In Fig. 5.2 (a), we showed the multi-slice reconstructions without using the K-L divergence penalty (i.e., $\lambda_{22} = 0$).

The diastole and systole phases for each slice are exhibited. At the bottom of the figure, we show the plots of the latent vectors for each slice. From the plots of the latent vectors in Fig. 5.2 (a)-bottom row, we see that the distribution of the latent vectors for each slice are different. For instance, the vectors are well distributed for slice no. 3, while they are more concentrated for slice no. 6. When the generator is fed with the latent vectors corresponding to slice no. 3, the images of other slices are poor, as shown in Fig. 5.2 (a)-top rows. In Fig. 5.2 (b), we show the multi-slice reconstructions by adding the K-L divergence penalty. From the plots of the latent vectors, we see that the latent vectors for each slice will have similar distributions. When the generator is excited with the latent vectors of slice no. 3, the reconstructions of all the slices have higher SER.



(a) Reconstructions without K-L divergence penalty



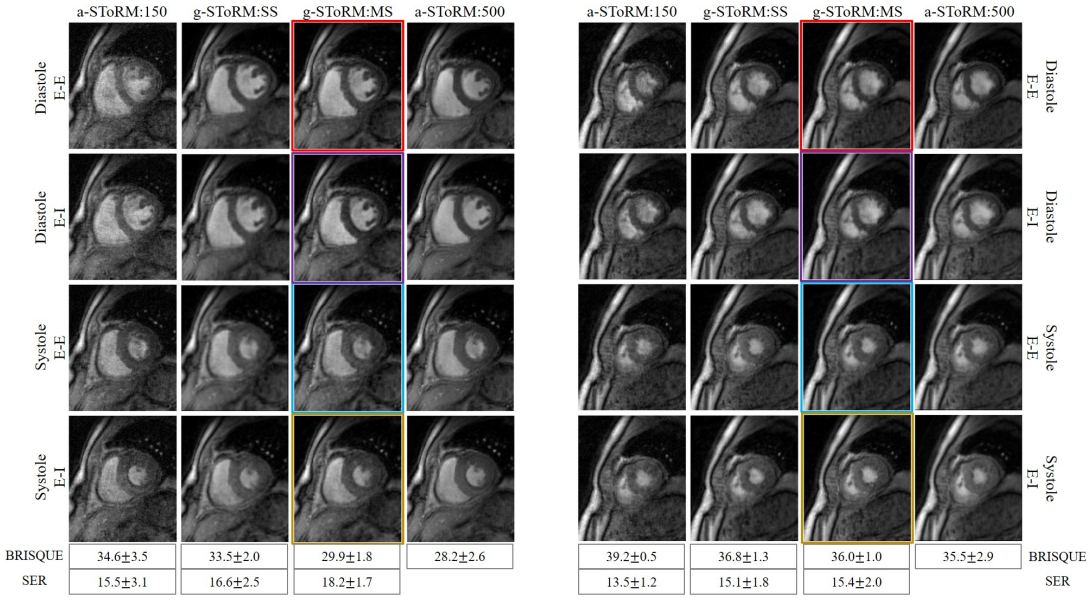
(b) Reconstructions with K-L divergence penalty

Figure 5.2. Illustration of the impact of the K-L divergence penalty. We use four slices (slices 3-6) in the first dataset from the healthy volunteer to generate the results. In (a), we show the multi-slice reconstructions without using the K-L divergence penalty. The latent vectors corresponding to slice 3, which is shown in the plot at the bottom of slice 3, are fed into the generator to obtain the multi-slice reconstructions. Since the latent vectors in this case have different distributions, the reconstructions of slices 4, 5, and 6 are of bad image quality. In (b), we show the multi-slice reconstructions using the K-L divergence penalty. We feed the latent vectors corresponding to slice 3 into the generator. From the plots of the latent vectors, which are shown at the bottom of Fig. (b), we can see that the latent vectors of each slice have the same distribution, hence resulting in good reconstruction.

5.3.2 Comparisons with current manifold methods

The results and comparisons for the datasets from the healthy volunteers are shown in Fig. 5.3 and Fig. 5.4. The results show that the generative manifold approach is able to reduce noise and alias artifacts compared to analysis SToRM. We attribute the improved performance to spatial regularization offered by the CNN generator, which is absent in the analysis SToRM formulation. Furthermore, we note that, unlike the analysis scheme, the proposed approach does not use k-space navigators to estimate the motion states; the latent variables are estimated from the measured k-space data itself. In addition, we see that the multi-slice reconstruction capitalizes on the redundancy between the slices compared to the single-slice generative SToRM reconstructions, offering improved performance and around 2dB improvement in performance.

The results and comparisons for the COPD dataset are shown in Fig. 5.5. The relatively irregular respiration makes this a challenging dataset to recover. From Fig. 5.5, we see that the competing methods have difficulty capturing the left ventricular (LV) boundaries in the diastole phase. By contrast, the multi-slice reconstructions can offer improved reconstruction and reduced blurring.

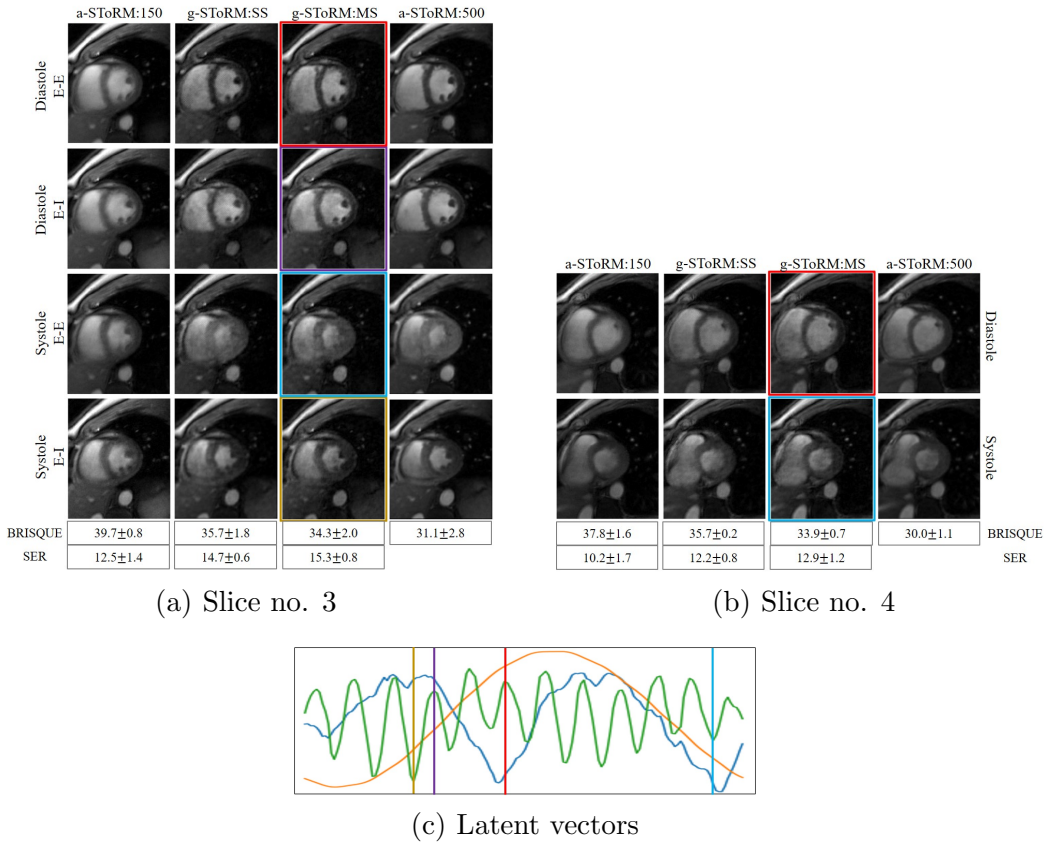


(a) Slice no. 3

(b) Slice no. 6

(c) Latent vectors

Figure 5.3. Illustration of the framework of the proposed scheme and comparison with existing methods. The experiments are based on the first dataset from the healthy volunteer, and 8 slices are used. We compare the proposed multi-slice generative manifold approach with the analysis SToRM and single-slice generative SToRM approaches. We use the SToRM reconstructions from the data of 500 frames (a-SToRM:500) as the reference for quantitative comparison. For the comparisons, we use the data of 150 frames for the reconstruction. From the reported average SER, shown at the bottom of figures (a) and (b), one can see that the proposed multi-slice generative manifold approach offers better reconstructions than the competing methods. We also plot the latent variables of 150 frames in time series for the proposed method. In this experiment, we feed the latent vectors corresponding to slice 8 to generate the multi-slice reconstruction. We showed four different phases for two different slices that are reconstructed in the time series: systole in end-expiration (E-E), systole in end-inspiration (E-I), diastole in E-E and diastole in E-I. The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors.

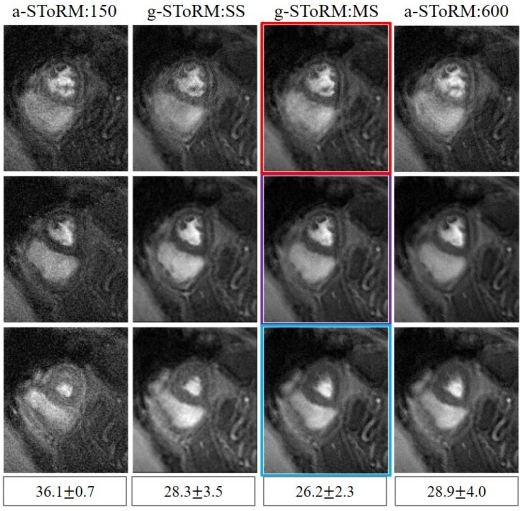


(a) Slice no. 3

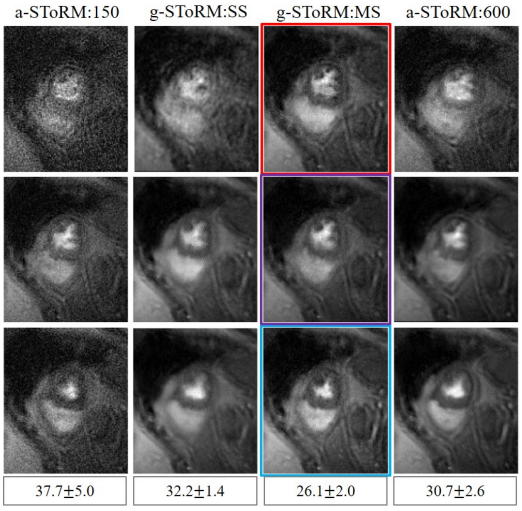
(b) Slice no. 4

(c) Latent vectors

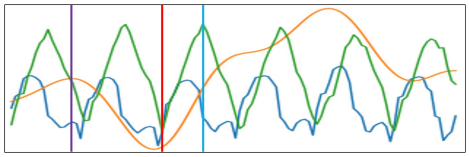
Figure 5.4. Illustration of the framework of the proposed scheme and comparison with existing methods. The experiments are based on the second dataset from the healthy volunteer, and 5 slices are used. We compare the proposed multi-slice generative manifold approach with the analysis SToRM and single-slice generative SToRM approaches. We use the SToRM reconstructions from the data of 500 frames (a-SToRM:500) as the reference for quantitative comparison. For the comparisons, we use the data of 150 frames for the reconstruction. From the reported average SER, shown at the bottom of figures (a) and (b), one can see that the proposed multi-slice generative manifold approach offers better reconstructions than the competing methods. We also plot the latent variables of 150 frames in time series for the proposed method. The first three slices in this dataset have the liver appearing in the field of view, but it never appears in the last two slices. Therefore, it is hard to determine the respiratory phases for the last two slices. In this experiment, we feed the latent vectors corresponding to slice 2 to generate the multi-slice reconstruction. We showed four different phases for slice 3 that are reconstructed in the time series and two phases for slice 4. The latent vectors corresponding to the four different phases are indicated in the plot of the latent vectors.



(a) Slice no. 2



(b) Slice no. 3



(c) Latent vectors

Figure 5.5. Illustration of the framework of the proposed scheme and comparison with existing methods. The experiments are based on the COPD dataset. We compared the proposed multi-slice generative manifold approach with the analysis SToRM and single-slice generative SToRM approaches. For the comparisons, we use the data of 150 frames for the reconstruction. We also compare the results with the analysis SToRM reconstructions from the data of 600 frames (a-SToRM:600). The BRISQUE score is used for quantitative comparison. The numbers at the bottom of figures (a) and (b) are the average BRISQUE scores. From the reported BRISQUE scores, one can see that the proposed multi-slice generative manifold approach offers better perceptual image quality than the competing methods. We also plot the latent variables of 150 frames in time series for the proposed method. In this experiment, we feed the latent vectors corresponding to slice 2 to generate the multi-slice reconstruction. We showed three different phases for two different slices that are reconstructed in the time series: diastole (first row), systole (third row), and intermediate phase (second row). The latent vectors corresponding to the three different phases are indicated in the plot of the latent vectors.

5.4 Discussion & Conclusions

In this note, we introduce a generative manifold representation for the alignment and joint recovery of multi-slice dynamic MRI from highly undersampled measurements. The deep CNN generator, which maps low-dimensional latent variables to a smooth image manifold, is used to represent and recover the images from highly undersampled data. The key benefit of this approach over current deep learning methods is that it does not require fully sampled training data, which is difficult to acquire in the free-breathing setting. Unlike current manifold approaches that perform the independent recovery of the slices, the proposed approach jointly recovers the images from the undersampled k-t space data of all the slices, thus exploiting the inter-slice redundancies.

Our results show that the the joint recovery of the slices offers reduced blurring and reduction of artifacts compared to g-SToRM:SS. Similarly, the use of the CNN generator offers implicit spatial regularization, resulting in improved recovery over a-SToRM:SS. The g-SToRM:MS framework is able to provide results that are comparable to the classical a-SToRM:SS approach with three fold less acquisition time.

CHAPTER 6

SUMMARY

In this thesis, we consider the novel union of surfaces model for signal processing. We first build the foundation of the union of surfaces model, including the mathematical background of the union of surfaces model, the relation between the union of surfaces model and the widely used union of subspaces model, and the recovery of union of surfaces from incomplete and noisy samples. We propose a kernel low-rank algorithm for the recovery the union of surfaces in the first part of this thesis.

Then we study the recovery of the functions that are living on the union of surfaces. We show when can we exactly learn and recover a function that lives on a surface, from few input-output examples. Based on which, we give the explanation of the good performance of imaging algorithms that use manifold structure. In this part, we focus on surface recovery in high-dimensional spaces with application to machine learning and learning surfaces of patches and images. We also link the computational structure of the proposed function learning algorithm to neural networks in the second part of this thesis.

In the third part of this thesis, we apply the proposed union of surfaces model to computational images. We consider the reconstruction of free-breathing and ungated cardiac MRI for the application. We also extend the results in the third part of the thesis to the multi-slice MRI setting in the fourth part of this thesis.

REFERENCES

[1] Aim@shape, digital shape workbench. `http://www.infra-visionair.eu/`.

[2] GOLD. `https://goldcopd.org/`.

[3] Hemant K Aggarwal, Merry P Mani, and Mathews Jacob. Modl: Model-based deep learning architecture for inverse problems. *IEEE transactions on medical imaging*, 38(2):394–405, 2018.

[4] A. Aghasi, M. Kilmer, and E. L. Miller. Parametric level set methods for inverse problems. *SIAM J. Imaging Sci.*, 4(2):618–650, 2011.

[5] Abdul Haseeb Ahmed, Ruixi Zhou, Yang Yang, Prashant Nagpal, Michael Salerno, and Mathews Jacob. Free-breathing and ungated dynamic mri using navigator-less spiral storm. *IEEE Transactions on Medical Imaging*, 2020.

[6] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[7] M.F. Atiyah and I.G. MacDonald. *Introduction To Commutative Algebra*. Addison-Wesley series in mathematics. Avalon Publishing, 1994.

[8] Leon Axel and Ricardo Otazo. Accelerated mri for the assessment of cardiac function. *The British journal of radiology*, 89(1063):20150655, 2016.

[9] A. Badoual, D. Schmitter, V. Uhlmann, and M. Unser. Multiresolution subdivision snakes. *IEEE Trans. Image Process.*, 26(3):1188–1201, 2017.

[10] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.*, 7(2006):2399–2434, 2006.

[11] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(Nov):2399–2434, 2006.

[12] O. Bernard, D. Friboulet, P. Thévenaz, and M. Unser. Variational B-spline level-set: a linear filtering approach for fast deformable model evolution. *IEEE Trans. Image Process.*, 18(6):1179–1191, 2009.

[13] Olivier Bernard, Denis Friboulet, Philippe Thévenaz, and Michael Unser. Variational b-spline level-set: a linear filtering approach for fast deformable model evolution. *IEEE Transactions on Image Processing*, 18(6):1179–1191, 2009.

[14] T. Blu, P.-L. Dragotti, M. Vetterli, P. Marziliano, and L. Coulot. Sparse sampling of signal innovations. *IEEE Signal Process. Mag.*, 25(2):31–40, 2008.

[15] A. Bora, E. Price, and A. G. Dimakis. Ambientgan: Generative models from lossy measurements. In *International Conference on Learning Representations*, 2018.

[16] M. Botsch, M. Pauly, L. Kobbelt, P. Alliez, B. Lévy, S. Bischoff, and C. Rössl. Geometric modeling based on polygonal meshes. *ACM SIGGRAPH 2007 courses - SIGGRAPH '07*, 2007.

[17] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. Non-local means denoising. *Image Processing On Line*, 1:208–212, 2011.

[18] M. Burger and S. Osher. A survey on level set methods for inverse problems and optimal design. *Eur. J. Appl. Math.*, 16(02):263–301, 2005.

[19] A. Bustin, N. Fuin, R. M. Botnar, and C. Prieto. From compressed-sensing to artificial intelligence-based cardiac mri reconstruction. *Frontiers in Cardiovascular Medicine*, 7:17, 2020.

[20] Aurelian Bustin, Niccolo Fuin, Rene M Botnar, and Claudia Prieto. From compressed-sensing to artificial intelligence-based cardiac mri reconstruction. *Frontiers in Cardiovascular Medicine*, 7(17), 2020.

[21] E. J. Candès and C. Fernandez-Granda. Towards a mathematical theory of super-resolution. *Commun. Pure Appl. Math.*, 67(6):906–956, 2014.

[22] Xin Chen, Muhammad Usman, Christian F. Baumgartner, Daniel R. Balfour, Paul K. Marsden, Andrew J. Reader, Claudia Prieto, and Andrew P. King. High-Resolution Self-Gated Dynamic Abdominal MRI Using Manifold Alignment. *IEEE Transactions on Medical Imaging*, 36(4):960–971, 2017.

[23] Youngmin Cho and Lawrence K Saul. Kernel methods for deep learning. In *Advances in neural information processing systems*, pages 342–350, 2009.

[24] Anthony G Christodoulou, Jaime L Shaw, Christopher Nguyen, Qi Yang, Yibin Xie, Nan Wang, and Debiao Li. Magnetic resonance multitasking for motion-resolved quantitative cardiovascular imaging. *Nature biomedical engineering*, 2(4):215–226, 2018.

[25] K. Crane, U. Pinkall, and P. Schröder. Robust fairing via conformal curvature flow. *ACM Trans. Graph.*, 32(4):1–10, 2013.

[26] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising with block-matching and 3D filtering. In Nasser M. Nasrabadi, Syed A. Rizvi, Edward R. Dougherty, Jaakko T. Astola, and Karen O. Egiazarian, editors, *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, pages 354 – 365. International Society for Optics and Photonics, SPIE, 2006.

[27] S. UH Dar et al. Image synthesis in multi-contrast mri with conditional generative adversarial networks. *IEEE transactions on medical imaging*, 38(10):2375–2388, 2019.

[28] S. UH Dar et al. Prior-guided image reconstruction for accelerated multi-contrast mri via generative adversarial networks. *IEEE Journal of Selected Topics in Signal Processing*, 14(6):1072–1087, 2020.

[29] S. UH Dar, M. Özbey, A. B. Çatlı, and T. Çukur. A transfer-learning approach for accelerated mri using deep neural networks. *Magnetic resonance in medicine*, 84(2):663–685, 2020.

[30] G. Dardikman-Yoffe and Y. C. Eldar. Learned sparcom: Unfolded deep super-resolution microscopy. *arXiv preprint arXiv:2004.09270*, 2020.

[31] R. Delgado-gonzalo, V. Uhlmann, D. Schmitter, and M. Unser. Snakes on a Plane: A perfect snap for bioimage analysis. *IEEE Signal Process. Mag.*, 32(1):41–48, 2015.

[32] Zixin Deng, Jianing Pang, Wensha Yang, Yong Yue, Behzad Sharif, Richard Tuli, Debiao Li, Benedick Fraass, and Zhaoyang Fan. Four-dimensional mri using three-dimensional radial sampling with respiratory self-gating to characterize temporal phase-resolved respiratory motion in the abdomen. *Magnetic resonance in medicine*, 75(4):1574–1585, 2016.

[33] P. L. Dragotti, M. Vetterli, and T. Blu. Sampling moments and reconstructing signals of finite rate of innovation: Shannon meets Strang–Fix. *IEEE Trans. Signal Process.*, 55(5):1741–1757, 2007.

[34] John Duchi. Derivations for linear algebra and optimization. *Berkeley, California*, 3(1):2325–5870, 2007.

[35] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006.

[36] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(11):2765–2781, 2013.

[37] M. Fatemi, A. Amini, and M. Vetterli. Sampling and reconstruction of shapes with algebraic boundaries. *IEEE Trans. Signal Process.*, 64(22):5807–5818, 2016.

[38] Herbert Federer. *Geometric measure theory*. Springer, 2014.

[39] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.

[40] Li Feng, Leon Axel, Hersh Chandarana, Kai Tobias Block, Daniel K Sodickson, and Ricardo Otazo. Xd-grasp: golden-angle radial mri with reconstruction of extra motion-state dimensions using compressed sensing. *Magnetic resonance in medicine*, 75(2):775–788, 2016.

[41] Li Feng, Robert Grimm, Kai Tobias Block, Hersh Chandarana, Sungheon Kim, Jian Xu, Leon Axel, Daniel K Sodickson, and Ricardo Otazo. Golden-angle radial sparse parallel mri: combination of compressed sensing, parallel imaging, and golden-angle radial sampling for fast and flexible dynamic volumetric mri. *Magnetic resonance in medicine*, 72(3):707–717, 2014.

[42] M. Fornasier, H. Rauhut, and R. Ward. Low-rank matrix recovery via iteratively reweighted least squares minimization. *SIAM J. Optim.*, 21(4):1614–1640, 2011.

[43] Vittorio Gallese. The roots of empathy: the shared manifold hypothesis and the neural basis of intersubjectivity. *Psychopathology*, 36(4):171–180, 2003.

[44] Kfir Gedalyahu and Yonina C Eldar. Time-delay estimation from low-rate samples: A union of subspaces approach. *IEEE Transactions on Signal Processing*, 58(6):3017–3031, 2010.

[45] I. Goodfellow et al. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[46] Robert Clifford Gunning and Hugo Rossi. *Analytic functions of several complex variables*, volume 368. American Mathematical Soc., 2009.

[47] J. P. Haldar. Low-Rank modeling of local -space neighborhoods (LORAKS) for constrained MRI. *IEEE Trans. Med. Imaging*, 33(3):668–681, 2014.

[48] Y. Han et al. Deep learning with domain adaptation for accelerated projection-reconstruction mr. *Magnetic resonance in medicine*, 80(3):1189–1205, 2018.

[49] Trevor Hastie, Robert Tibshirani, Jerome Friedman, and James Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.

[50] Aaron S Jackson, Adrian Bulat, Vasileios Argyriou, and Georgios Tzimiropoulos. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1031–1039, 2017.

[51] M. Jacob, T. Blu, and M. Unser. Efficient energies and algorithms for parametric snakes. *IEEE Trans. Image Process.*, 13(9):1231–1244, 2004.

[52] Mathews Jacob, Thierry Blu, and Michael Unser. Efficient energies and algorithms for parametric snakes. *IEEE transactions on image processing*, 13(9):1231–1244, 2004.

[53] R. Jain, R. Kasturi, and B. G. Schunck. Curves and surfaces. *Mach. Vis.*, pages 365–405, 1995.

[54] K. H. Jin, H. Gupta, J. Yerly, M. Stuber, and M. Unser. Time-dependent deep image prior for dynamic mri. *arXiv preprint arXiv:1910.01684*, 2019.

[55] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017.

[56] Kyong Hwan Jin, Harshit Gupta, Jérôme Yerly, Matthias Stuber, and Michael Unser. Time-dependent deep image prior for dynamic MRI, 2019.

[57] Tomoyuki Kido, Teruhito Kido, Masashi Nakamura, Kouki Watanabe, Michaela Schmidt, Christoph Forman, and Teruhito Mochizuki. Compressed sensing real-time cine cardiovascular magnetic resonance: accurate assessment of left ventricular function in a single-breath-hold. *Journal of Cardiovascular Magnetic Resonance*, 18(1):1–11, 2016.

[58] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[59] Christos Koulamas and George J Kyparisis. Single-machine and two-machine flowshop scheduling with general learning functions. *European Journal of Operational Research*, 178(2):402–407, 2007.

[60] Thomas Küstner, Aurelien Bustin, Olivier Jaubert, Reza Hajhosseiny, Pier Giorgio Masci, Radhouene Neji, René Botnar, and Claudia Prieto. Isotropic 3D Cartesian single breath-hold CINE MRI with multi-bin patch-based low-rank reconstruction. *Magnetic Resonance in Medicine*, 84(4), 2020.

[61] Thomas Küstner, Niccolo Fuin, Kerstin Hammernik, Aurelien Bustin, Haikun Qi, Reza Hajhosseiny, Pier Giorgio Masci, Radhouene Neji, Daniel Rueckert, René M Botnar, et al. Cinenet: deep learning-based 3d cardiac cine mri reconstruction with multi-coil complex-valued 4d spatio-temporal convolutions. *Scientific reports*, 10(1):1–13, 2020.

[62] Y. LeCun, C. Cortes, and C.J. Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

[63] Victor Lempitsky, Andrea Vedaldi, and Dmitry Ulyanov. Deep image prior. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9446–9454. IEEE, 2018.

[64] C. Li, C. Xu, C. Gui, and M. D. Fox. Distance regularized level set evolution and its application to image segmentation. *IEEE trans. image process.*, 19(12):3243–3254, 2010.

[65] Tao Li, Alexandre Krupa, and Christophe Collewet. A robust parametric active contour based on fourier descriptors. In *2011 18th IEEE International Conference on Image Processing*, pages 1037–1040. IEEE, 2011.

[66] Sajan Goud Lingala, Yue Hu, Edward DiBella, and Mathews Jacob. Accelerated dynamic mri exploiting sparsity and low-rank structure: kt slr. *IEEE transactions on medical imaging*, 30(5):1042–1054, 2011.

[67] B. F. Logan. Information in the zero crossings of bandpass signals. *Bell Syst. Tech. J.*, 56(4):487–510, 1977.

[68] Yue M Lu and Minh N Do. A theory for sampling signals from a union of subspaces. *IEEE transactions on signal processing*, 56(6):2334–2345, 2008.

[69] Michael Lustig, David Donoho, and John M Pauly. Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007.

[70] Julien Mairal, Piotr Koniusz, Zaid Harchaoui, and Cordelia Schmid. Convolutional kernel networks. In *Advances in neural information processing systems*, pages 2627–2635, 2014.

[71] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(7):710–732, 1992.

[72] S. Maymon and A. V. Oppenheim. Sinc interpolation of nonuniform samples. *IEEE Trans. Signal Process.*, 59(10):4745–4758, 2011.

[73] Moshe Mishali, Yonina C Eldar, and Asaf J Elron. Xampling: Signal acquisition and processing in union of subspaces. *IEEE Transactions on Signal Processing*, 59(10):4719–4734, 2011.

[74] A. Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012.

[75] K. Mohan and M. Fazel. Iterative reweighted algorithms for matrix rank minimization. *J. Mach. Learn. Res.*, 13(Nov):3441–3473, 2012.

[76] Y. Q. Mohsin, G. Ongie, and M. Jacob. Iterative shrinkage algorithm for patch-smoothness regularized medical image recovery. *IEEE Trans. Med. Imaging*, 34(12):2417–2428, 2015.

[77] Yasir Q Mohsin, Sajan Goud Lingala, Edward DiBella, and Mathews Jacob. Accelerated dynamic mri using patch regularization for implicit motion compensation. *Magnetic resonance in medicine*, 77(3):1238–1248, 2017.

[78] V. Monga, Y. Li, and Y. C. Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *arXiv preprint arXiv:1912.10557*, 2019.

[79] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Netw.*, 12(2):181–201, 2001.

[80] Ukash Nakarmi, Konstantinos Slavakis, and Leslie Ying. Mls: Joint manifold-learning and sparsity-aware framework for highly accelerated dynamic magnetic resonance imaging. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 1213–1216. IEEE, 2018.

[81] Ukash Nakarmi, Yanhua Wang, Jingyuan Lyu, Dong Liang, and Leslie Ying. A kernel-based low-rank (klr) model for low-dimensional manifold recovery in highly accelerated dynamic mri. *IEEE transactions on medical imaging*, 36(11):2297–2307, 2017.

[82] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa. Laplacian mesh optimization. *Siggraph*, pages 381–389, 2006.

[83] G. Ongie, S. Biswas, and M. Jacob. Convex recovery of continuous domain piecewise constant images from nonuniform Fourier samples. *IEEE Trans. Signal Process.*, 66(1):236–250, 2017.

[84] G. Ongie and M. Jacob. Recovery of piecewise smooth images from few fourier samples. In *2015 International Conference on Sampling Theory and Applications (SampTA)*, pages 543–547. IEEE, 2015.

[85] G. Ongie and M. Jacob. Off-the-grid recovery of piecewise constant images from few fourier samples. *SIAM J. Imaging Sci.*, 9(3):1004–1041, 2016.

[86] G. Ongie and M. Jacob. A fast algorithm for convolutional structured low-rank matrix recovery. *IEEE Trans. Comput. Imaging*, 3(4):535–550, 2017.

[87] G. Ongie, R. Willett, R. D. Nowak, and L. Balzano. Algebraic variety models for high-rank matrix completion. In *Proc. 34th Int. Conf. Mach. Learn.*, pages 2691–2700, 2017.

[88] Greg Ongie and Mathews Jacob. Off-the-grid recovery of piecewise constant images from few fourier samples. *SIAM Journal on Imaging Sciences*, 9(3):1004–1041, 2016.

[89] Greg Ongie, Rebecca Willett, Robert D Nowak, and Laura Balzano. Algebraic variety models for high-rank matrix completion. *arXiv preprint arXiv:1703.09631*, 2017.

[90] Stanley Osher and Ronald P Fedkiw. Level set methods: an overview and some recent results. *Journal of Computational physics*, 169(2):463–502, 2001.

[91] H. Pan, T. Blu, and P. L. Dragotti. Sampling curves with finite rate of innovation. *IEEE Trans. Signal Process.*, 62(2):458–471, 2014.

[92] Hanjie Pan, Thierry Blu, and Pier Luigi Dragotti. Sampling curves with finite rate of innovation. *IEEE Transactions on Signal Processing*, 62(2):458–471, 2013.

[93] Gabriel Peyré and Stéphane Mallat. Surface compression with geometric bandelets. *ACM Transactions on Graphics (TOG)*, 24(3):601–608, 2005.

[94] Gianluigi Pillonetto, Francesco Dinuzzo, Tianshi Chen, Giuseppe De Nicolao, and Lennart Ljung. Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3):657–682, 2014.

[95] S. Poddar and M. Jacob. Recovery of noisy points on band-limited surfaces: kernel methods re-explained. *arXiv preprint arXiv:1801.00890*, 2018.

[96] S. Poddar and M. Jacob. Recovery of point clouds on surfaces: Application to image reconstruction. In *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*, pages 1272–1275. IEEE, 2018.

[97] Sunrita Poddar and Mathews Jacob. Dynamic mri using smoothness regularization on manifolds (storm). *IEEE transactions on medical imaging*, 35(4):1106–1115, 2015.

[98] Sunrita Poddar and Mathews Jacob. Recovery of noisy points on bandlimited surfaces: Kernel methods re-explained. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4024 – 4028. IEEE, 2018.

[99] Sunrita Poddar and Mathews Jacob. Recovery of point clouds on surfaces: Application to image reconstruction. In *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*, pages 1272–1275. IEEE, 2018.

[100] Sunrita Poddar, Yasir Q Mohsin, Deidra Ansah, Bijoy Thattaliyath, Ravi Ashwath, and Mathews Jacob. Manifold recovery using kernel low-rank regularization: application to dynamic imaging. *IEEE Transactions on Computational Imaging*, 5(3):478–491, 2019.

[101] Daniel Potts and Gabriele Steidl. Fourier reconstruction of functions from their nonstandard sampled radon transform. *Journal of Fourier Analysis and Applications*, 8(6):513–534, 2002.

[102] A. Pramanik, H. K. Aggarwal, and M. Jacob. Deep generalization of structured low-rank algorithms (deep-slr). *IEEE Transactions on Medical Imaging*, 2020.

[103] C. Prieto et al. Highly efficient respiratory motion compensated free-breathing coronary mra using golden-step cartesian acquisition. *Journal of Magnetic Resonance Imaging*, 41(3):738–746, 2015.

[104] Chen Qin, Jo Schlemper, Jose Caballero, Anthony N. Price, Joseph V. Hajnal, and Daniel Rueckert. Convolutional Recurrent Neural Networks for Dynamic MR Image Reconstruction. *IEEE Transactions on Medical Imaging*, 38(1):280–290, jan 2019.

[105] Ali Rahimi and Benjamin Recht. Weighted sums of random kitchen sinks: replacing minimization with randomization in learning. In *Nips*, pages 1313–1320. Citeseer, 2008.

[106] Michael Rauth and Thomas Strohmer. Smooth approximation of potential fields from noisy scattered data. *Geophysics*, 63(1):85–94, 1998.

[107] Saiprasad Ravishankar and Yoram Bresler. Learning doubly sparse transforms for images. *IEEE Transactions on Image Processing*, 22(12):4598–4612, 2013.

[108] Sebastian Rosenzweig, Nick Scholand, H Christian M Holme, and Martin Uecker. Cardiac and respiratory self-gating in radial mri using an adapted singular spectrum analysis (ssa-fary). *IEEE transactions on medical imaging*, 39(10):3029–3041, 2020.

[109] Mikael Rousson and Nikos Paragios. Shape priors for level set representations. In *European Conference on Computer Vision*, pages 78–92. Springer, 2002.

[110] Paul Sajda, Andrew Laine, and Yehoshua Zeevi. Multi-resolution and wavelet representations for identifying signatures of disease. *Disease markers*, 18(5, 6):339–363, 2002.

[111] T. Sanchez et al. Scalable learning-based sampling optimization for compressive dynamic mri. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8584–8588. IEEE, 2020.

[112] Christopher M Sandino, Peng Lai, Shreyas S Vasanawala, and Joseph Y Cheng. Accelerating cardiac cine mri using a deep learning-based espirit reconstruction. *Magnetic Resonance in Medicine*, 85(1):152–167, 2021.

[113] G. Schiebinger, E. Robeva, and B. Recht. Superresolution without separation. In *Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP), 2015 IEEE 6th Int. Work.*, pages 45–48. IEEE, 2015.

[114] B. Schölkopf and A. J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[115] Florian Schroff, Antonio Criminisi, and Andrew Zisserman. Object class segmentation using random forests. In *BMVC*, pages 1–10, 2008.

[116] I. R. Shafarevich. *Basic algebraic geometry. 1. Varieties in projective space (Thrid edition)*. Springer-Verlang, 2013.

[117] Hang Shao, Abhishek Kumar, and P Thomas Fletcher. The riemannian geometry of deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 315–323, 2018.

[118] Gaurav N Shetty, Konstantinos Slavakis, Abhishek Bose, Ukash Nakarmi, Gesualdo Scutari, and Leslie Ying. Bi-linear modeling of data manifolds for dynamic-mri recovery. *IEEE transactions on medical imaging*, 39(3):688–702, 2019.

[119] P. Shukla and P. L. Dragotti. Sampling schemes for multidimensional signals with finite rate of innovation. *IEEE Trans. Signal Process.*, 55(7):3670–3686, 2007.

[120] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Process. Mag.*, 30(3):83–98, 2013.

[121] K. Siddiqi, Y. Lauziere, and S. W. Tannenbaum, A.and Zucker. Area and length minimizing flows for shape segmentation. *IEEE trans. image process.*, 7(3):433–443, 1998.

[122] Alexander J Smola and Risi Kondor. Kernels and regularization on graphs. In *Learning theory and kernel machines*, pages 144–158. Springer, 2003.

[123] Tor Sørevik and Morten A Nome. Trigonometric interpolation on lattice grids. *BIT Numerical Mathematics*, 56(1):341–356, 2016.

[124] Thomas Strohmer. Computationally attractive reconstruction of bandlimited images from irregular samples. *IEEE Transactions on image processing*, 6(4):540–548, 1997.

[125] Thomas Strohmer, Thomas Binder, and M Sussner. How to recover smooth object boundaries in noisy medical images. In *Proceedings of 3rd IEEE International Conference on Image Processing*, volume 1, pages 331–334. IEEE, 1996.

[126] Chunwei Tian, Yong Xu, Lunke Fei, Junqian Wang, Jie Wen, and Nan Luo. Enhanced cnn for image denoising. *CAAI Transactions on Intelligence Technology*, 4(1):17–23, 2019.

[127] Jeffrey Tsao, Peter Boesiger, and Klaas P Pruessmann. k-t blast and k-t sense: dynamic mri with high frame rate exploiting spatiotemporal correlations. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 50(5):1031–1042, 2003.

[128] G. Turk and J. F. O'Brien. Shape transformation using variational implicit functions. *Proc. 26th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH 99*, 33(Annual Conference Series):335–342, 1999.

[129] Martin Uecker, Peng Lai, Mark J Murphy, Patrick Virtue, Michael Elad, John M Pauly, Shreyas S Vasanawala, and Michael Lustig. Espirit—an eigenvalue approach to autocalibrating parallel mri: where sense meets grappa. *Magnetic resonance in medicine*, 71(3):990–1001, 2014.

[130] V. Uhlmann, J. Fageot, and M. Unser. Hermite snakes with control of tangents. *IEEE Trans. Image Process.*, 25(6):2803–2816, 2016.

[131] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018.

[132] Muhammad Usman, David Atkinson, Christoph Kolbitsch, Tobias Schaeffter, and Claudia Prieto. Manifold learning based ECG-free free-breathing cardiac CINE MRI. *Journal of Magnetic Resonance Imaging*, 41(6), 2015.

[133] D. Varga, A. Csiszárik, and Z. Zombori. Gradient regularization improves accuracy of discriminative models. *arXiv preprint arXiv:1712.09936*, 2017.

[134] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Trans. Signal Process.*, 50(6):1417–1428, 2002.

[135] D. O. Walsh, A. F. Gmitro, and M. W. Marcellin. Adaptive reconstruction of phased array mr imagery. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 43(5):682–690, 2000.

[136] G. Wang. A perspective on deep imaging. *IEEE Access*, 4:8914–8924, 2016.

[137] G. Wang, M. Kalra, and C. G. Orton. Machine learning will transform radiology significantly within the next 5 years. *Medical Physics*, 44(6):2041–2044, 2017.

[138] S. Wang et al. Accelerating magnetic resonance imaging via deep learning. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 514–517. IEEE, 2016.

[139] S. Wang et al. Deepcomplexmri: Exploiting deep residual network for fast parallel mr imaging with complex convolution. *Magnetic Resonance Imaging*, 68:136–147, 2020.

[140] Shengyin Wang and Michael Yu Wang. Radial basis functions and level set method for structural topology optimization. *International journal for numerical methods in engineering*, 65(12):2060–2090, 2006.

[141] Tianchen Wang, Xiaowei Xu, Jinjun Xiong, Qianjun Jia, Haiyun Yuan, Meiping Huang, Jian Zhuang, and Yiyu Shi. Ica-unet: Ica inspired statistical unet for real-time 3d cardiac cine mri segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 447–457. Springer, 2020.

[142] Ferenc Weisz. Summability of multi-dimensional trigonometric fourier series. *Surv. Approx. Theory*, 7:1–179, 2012.

[143] Zhili Yang and Mathews Jacob. Nonlocal regularization of inverse problems: a unified variational framework. *IEEE Transactions on Image Processing*, 22(8):3192–3203, 2012.

[144] J. C. Ye, Y. Han, and E. Cha. Deep convolutional framelets: A general deep learning framework for inverse problems. *SIAM Journal on Imaging Sciences*, 11(2):991–1048, 2018.

[145] Dong Yu, Kaisheng Yao, Hang Su, Gang Li, and Frank Seide. Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7893–7897. IEEE, 2013.

[146] M. Yurt, S. UH Dar, A. Erdem, E. Erdem, and T. Çukur. mustgan: Multi-stream generative adversarial networks for mr image synthesis. *arXiv preprint arXiv:1909.11504*, 2019.

[147] A. Zakhor and A. V Oppenheim. Reconstruction ot two-dimensional signals from level crossings. *Proc. IEEE*, 78(1):31–55, 1990.

[148] J. Zeng, G. Cheung, M. Ng, J. Pang, and C. Yang. 3d point cloud denoising using graph laplacian regularization of a low dimensional manifold model. *arXiv preprint arXiv:1803.07252*, 2018.

[149] Jun Zhang, Gilbert G Walter, Yubo Miao, and Wan Ngai Wayne Lee. Wavelet neural networks for function learning. *IEEE transactions on Signal Processing*, 43(6):1485–1497, 1995.

[150] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

[151] R. Zhou et al. Free-breathing cine imaging with motion-corrected reconstruction at 3t using spiral acquisition with respiratory correction and cardiac self-gating (sparcs). *Magnetic resonance in medicine*, 82(2):706–720, 2019.

[152] Ruixi Zhou, Yang Yang, Roshin C. Mathew, John P. Mugler, Daniel S. Weller, Christopher M. Kramer, Abdul Haseeb Ahmed, Mathews Jacob, and Michael Salerno. Free-breathing cine imaging with motion-corrected reconstruction at 3T using SPiral Acquisition with Respiratory correction and Cardiac Self-gating (SPARCS). *Magnetic Resonance in Medicine*, 2019.

[153] Q. Zou and M. Jacob. Recovery of surfaces and functions in high dimensions: sampling theory and links to neural networks. *SIAM Journal on Imaging Sciences*, in press.

[154] Qing Zou, Abdul Haseeb Ahmed, Prashant Nagpal, Stanley Kruger, and Mathews Jacob. Dynamic imaging using deep generative storm (gen-storm) model. *IEEE transactions on medical imaging*, 2021.

[155] Qing Zou, Sunrita Poddar, and Mathews Jacob. Sampling of planar curves: Theory and fast algorithms. *IEEE Transactions on Signal Processing*, 67(24):6455–6467, 2019.